

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«Харьковский политехнический институт»

В. П. Северин

**МЕТОДЫ
МНОГОМЕРНОЙ БЕЗУСЛОВНОЙ
МИНИМИЗАЦИИ**

Учебное пособие
по курсу «Методы оптимизации»

Для студентов направлений 6.040302 «Информатика»,
6.040303 «Системный анализ»

Утверждено
редакционно-издательским
советом НТУ «ХПИ»,
протокол № 2 от 06.12.2012 г.

Харьков
НТУ «ХПИ»
2013

УДК 519.85(075)

ББК 22.18я73

С 28

Рецензенты:

И. Ф. Домнин, д-р техн. наук, проф., Институт ионосферы
НАН Украины;

Н. В. Ткачук, д-р техн. наук, проф., Национальный технический
университет «Харьковский политехнический институт»

Розглянуті теорія і методи багатовимірної безумовної мінімізації. Викладена теорія необхідних і достатніх умов екстремуму функції багатьох змінних. Обґрунтовані формули та алгоритми більш ніж десяти основних методів безумовної мінімізації. Наведені завдання для лабораторних робіт.

Призначено для студентів технічних спеціальностей.

Северин В. П.

С 28 Методы многомерной безусловной минимизации : учеб. пособие по курсу «Методы оптимизации» / В. П. Северин. – Х. : НТУ «ХПИ», 2013. – 160 с. – На русск. яз.

ISBN 978-617-05-077-9

Рассмотрены теория и методы многомерной безусловной минимизации. Изложена теория необходимых и достаточных условий экстремума функции многих переменных. Обоснованы формулы и алгоритмы более десяти основных методов безусловной минимизации. Даны задания для лабораторных работ.

Предназначено для студентов технических специальностей.

Ил. 33. Табл. 5. Библиогр.: 23 назв.

УДК 519.85(075)

ББК 22.18я73

ISBN 978-617-05-077-9

© В.П. Северин, 2013

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1. ОСНОВЫ МЕТОДОВ МНОГОМЕРНОЙ ОПТИМИЗАЦИИ	7
1.1. Экстремум функции многих переменных	7
1.2. Условия оптимальности первого порядка	15
1.3. Условия оптимальности второго порядка	18
1.4. Метод циклического покоординатного спуска	23
1.5. Методы спуска	27
1.6. Метод наискорейшего спуска	30
1.7. Вычисление градиента	35
Лабораторная работа	37
Контрольные вопросы	41
2. МЕТОД НЬЮТОНА И ЕГО МОДИФИКАЦИИ	45
2.1. Метод Ньютона	45
2.2. Метод Ньютона с одномерным поиском	50
2.3. Метод Ньютона с направлением спуска	51
2.4. Метод Марквардта	55
2.5. Вычисление матрицы Гессе	62
Лабораторная работа	65
Контрольные вопросы	69
3. МЕТОДЫ СОПРЯЖЕННЫХ НАПРАВЛЕНИЙ	72
3.1. Свойства квадратичной функции	72
3.2. Сопряжённые векторы и их свойства	75
3.3. Теорема методов сопряжённых направлений	77
3.4. Метод Пауэлла	80
3.5. Методы сопряжённых градиентов	83
3.6. Метод Флетчера – Ривса	87
3.7. Метод Полака – Рибьера	92
Лабораторная работа	96
Контрольные вопросы	99

4. КВАЗИНЬЮТОНОВСКИЕ МЕТОДЫ	102
4.1. Основы квазиньютоновских методов	102
4.2. Метод Бройдена	105
4.3. Свойства метода Бройдена	108
4.4. Метод Девидона – Флетчера – Пауэлла	113
4.5. Свойства метода Девидона – Флетчера – Пауэлла	116
4.6. Метод Бройдена – Флетчера – Гольдфарба – Шанно	121
4.7. Модификация метода БФГШ	123
4.8. Сравнение методов безусловной оптимизации	126
Лабораторная работа	131
Контрольные вопросы	135
5. СПРАВОЧНЫЙ МАТЕРИАЛ	138
5.1. Правила дифференцирования	138
5.2. Элементы векторной алгебры	141
5.3. Матрицы и действия с ними	144
5.4. Формула Шермана – Моррисона и ее применение	148
5.5. Дифференцирование функций многих переменных	151
5.6. Формула Тейлора	154
5.7. Квадратичные формы	155
СПИСОК ЛИТЕРАТУРЫ	158

ВВЕДЕНИЕ

Методы оптимизации широко применяются в технике, физике, экономике и других областях. По теории оптимизации опубликовано достаточно много монографий и учебников с высоким математическим уровнем изложения материала. В то же время учебных пособий по методам оптимизации, рассчитанных на подготовку специалистов по техническим специальностям, явно недостаточно.

Предлагаемое учебное пособие учитывает специфику математической подготовки студентов высших технических учебных заведений. В основу пособия положен курс лекций по дисциплине «Методы оптимизации», который читается автором на протяжении ряда лет на факультете информатики и управления НТУ «ХПИ» для студентов специальностей «Информатика», «Социальная информатика», «Системный анализ и управление» и рассчитан на два семестра. Раздел «Методы безусловной минимизации» изучается в первом семестре. В предлагаемом втором модуле дисциплины «Методы оптимизации», состоящем из пяти разделов, рассмотрены теория и методы многомерной безусловной оптимизации.

В первом разделе рассматриваются вопросы, связанные с основными положениями теории многомерной безусловной оптимизации, и базовые методы. Даются определения минимума, максимума и экстремума функции многих переменных, формулируются и доказываются необходимые и достаточные условия экстремума. Описывается метод циклического покоординатного спуска. Рассматриваются общие формулы и свойства методов спуска. Обосновывается метод наискорейшего спуска, основанный на применении градиента целевой функции.

Второй раздел посвящен методам второго порядка для безусловной минимизации функции многих переменных, которые основаны на формуле Ньютона с использованием матрицы вторых частных производных функции – матрицы Гессе. Обосновываются метод Ньютона, метод Ньютона с одномерным поиском, метод Ньютона с заданием

направления спуска, метод Марквардта и его модификация.

В третьем разделе рассматриваются методы сопряженных направлений. Приводятся свойства квадратичной функции, вводится понятие сопряженных векторов, изучаются их свойства. Формулируется и доказывается теорема методов сопряженных направлений. Обосновывается метод Пауэлла, основанный на формировании сопряженных направлений. Формулируется и доказывается теорема методов сопряженных градиентов, даются общие свойства этих методов. Обосновываются методы Флетчера – Ривса и Полака – Рибьера, анализируются их достоинства и недостатки.

Четвертый раздел посвящен квазиньютоновским методам первого порядка, основанным на формуле Ньютона и на аппроксимации матрицы Гессе или обратной к ней матрицы. Приводятся теоретические основы квазиньютоновских методов. Обосновываются методы Бroyдена, Девидона – Флетчера – Пауэлла, Бroyдена – Флетчера – Гольдфарба – Шанно и модифицированный метод Бroyдена – Флетчера – Гольдфарба – Шанно. Не существует единственного самого эффективного метода для безусловной минимизации функции многих переменных. Дается сравнение скорости асимптотической сходимости методов многомерной безусловной минимизации и даны рекомендации выбора метода.

Для всех рассмотренных методов приведены алгоритмы и примеры, которые позволяют облегчить понимание методов и ускорить выполнение лабораторных работ. Даны задания для лабораторных работ.

Изучение теории и практики методов многомерной безусловной минимизации требует знания многих понятий и формул высшей математики. В пятом разделе обобщается основной справочный материал, необходимый для изучения методов многомерной оптимизации.

Особенностью пособия является последовательное изложение основных идей и методов многомерной безусловной минимизации.

Для усвоения материала достаточно владения стандартными курсами математического анализа и линейной алгебры.

1. ОСНОВЫ МЕТОДОВ МНОГОМЕРНОЙ ОПТИМИЗАЦИИ

Многомерная безусловная минимизация используется в методах условной оптимизации и во многом определяет их практическую эффективность. В данном разделе рассматриваются вопросы, связанные с основными положениями теории многомерной безусловной оптимизации, и базовые методы. Даются определения минимума, максимума и экстремума функции многих переменных, анализируются свойства экстремума, приводятся понятия выпуклой и вогнутой функций. Формулируются и доказываются необходимые и достаточные условия экстремума функции многих переменных. Описывается метод циклического покоординатного спуска. Рассматриваются общие формулы и свойства методов спуска. Обосновывается метод наискорейшего спуска и дается алгоритм метода. Рассматриваются формулы для вычисления градиента функции многих переменных. Приводится описание лабораторной работы по разработке подпрограмм для базовых методов безусловной минимизации.

1.1. Экстремум функции многих переменных

Методы безусловной оптимизации предназначены для вычисления экстремума целевой функции многих переменных $f(\mathbf{x})$, где $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ – вектор-столбец вещественных переменных, который также можно трактовать как точку n -мерного пространства $\mathbf{x} \in R^n$. Здесь и далее n – количество переменных, определяющее размерность вектора переменных параметров.

Точка \mathbf{x}^* называется *точкой локального минимума* функции $f(\mathbf{x})$, если существует такое $\delta > 0$, что $f(\mathbf{x}^*) \leq f(\mathbf{x})$ для всех \mathbf{x} , удовлетворяющих условию $\|\mathbf{x} - \mathbf{x}^*\| < \delta$.

Точка \mathbf{x}^* называется *точкой строгого локального минимума* функции $f(\mathbf{x})$, если существует такое $\delta > 0$, что $f(\mathbf{x}^*) < f(\mathbf{x})$ для всех \mathbf{x} , удовлетворяющих условиям $\mathbf{x} \neq \mathbf{x}^*$ и $\|\mathbf{x} - \mathbf{x}^*\| < \delta$.

Точка \mathbf{x}^* называется *точкой глобального минимума* функции $f(\mathbf{x})$, если $f(\mathbf{x}^*) \leq f(\mathbf{x})$ для всех $\mathbf{x} \in R^n$.

Точка \mathbf{x}^* называется *точкой строгого глобального минимума* функции $f(\mathbf{x})$, если $f(\mathbf{x}^*) < f(\mathbf{x})$ для всех $\mathbf{x} \in R^n$ и $\mathbf{x} \neq \mathbf{x}^*$.

Соответствующее значение функции $f^* = f(\mathbf{x}^*)$ называется ее *локальным минимумом, строгим локальным минимумом, глобальным минимумом, строгим глобальным минимумом*. Глобальный минимум является также и локальным минимумом.

Аналогично вводятся понятия *точек максимума* и *максимума* функции многих переменных. Минимум и максимум имеют общее название *экстремум*.

Для определенности методов оптимизации в них полагают, что необходимо найти точку минимума функции. Если для некоторой функции $g(\mathbf{x})$ необходимо найти максимум, то переходят к минимизации функции $f(\mathbf{x}) = -g(\mathbf{x})$.

Задача безусловной минимизации заключается в минимизации функции $f(\mathbf{x})$ при $\mathbf{x} \in R^n$, что представляется в виде:

$$\min f(\mathbf{x}), \quad \mathbf{x} \in R^n. \quad (1.1)$$

На вектор \mathbf{x} не накладывается никаких ограничений (условий), поэтому задача (1.1) минимизации функции $f(\mathbf{x})$ называется *задачей безусловной оптимизации* или *задачей оптимизации без ограничений*.

Функция, имеющая несколько минимумов или максимумов, называется *многоэкстремальной*.

Функция $f(\mathbf{x})$ называется *унимодальной*, если она имеет в про-

пространстве параметров R^n единственную точку минимума \mathbf{x}^* .

Решение задачи безусловной минимизации (1.1) представляется в виде:

$$\mathbf{x}^* = \arg \min f(\mathbf{x}), \quad \mathbf{x} \in R^n.$$

Функция $f(\mathbf{x})$ называется *выпуклой*, если

$$f(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) \leq \lambda f(\mathbf{x}_1) + (1-\lambda)f(\mathbf{x}_2) \quad (1.2)$$

для любых $\mathbf{x}_1, \mathbf{x}_2 \in R^n$ и любого $\lambda \in (0; 1)$. Функция $f(\mathbf{x})$ называется *строго выпуклой*, если неравенство (1.2) является строгим для всех различных $\mathbf{x}_1, \mathbf{x}_2$ и для любого $\lambda \in (0; 1)$. Функция $f(\mathbf{x})$ называется *вогнутой (строго вогнутой)*, если $-f(\mathbf{x})$ выпуклая (строго выпуклая). Выпуклые и вогнутые функции являются непрерывными функциями.

Дадим геометрическую интерпретацию выпуклой функции. Пусть $\mathbf{x}_1 = \mathbf{u}$ и $\mathbf{x}_2 = \mathbf{v}$ — две различные точки и определим точку $\mathbf{w} = \lambda \mathbf{u} + (1-\lambda)\mathbf{v}$ с $\lambda \in (0; 1)$. Заметим, что $y = f(\mathbf{w}) = f(\lambda \mathbf{u} + (1-\lambda)\mathbf{v})$ представляет значение функции в точке \mathbf{w} , и точка (\mathbf{w}, y) лежит на графике функции (рис. 1.1, а). Величина $z = \lambda f(\mathbf{u}) + (1-\lambda)f(\mathbf{v})$ дает взвешенную сумму $f(\mathbf{u})$ и $f(\mathbf{v})$, а точка (\mathbf{w}, z) лежит на графике секущей. Неравенство (1.2) примет вид $y \leq z$.

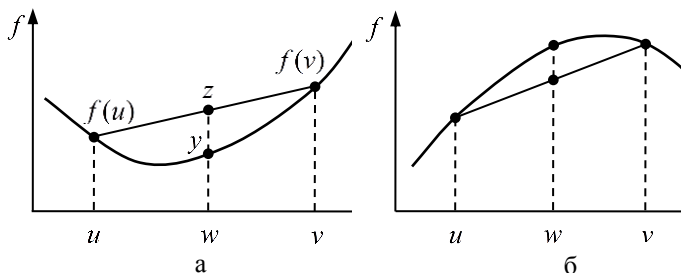


Рис. 1.1. Выпуклая и вогнутая функции

Таким образом, для выпуклой функции $f(\mathbf{x})$ значение в точках линии сегмента $\lambda \mathbf{u} + (1-\lambda)\mathbf{v}$ не больше высоты хорды, соединяющей точки $(\mathbf{u}, f(\mathbf{u}))$ и $(\mathbf{v}, f(\mathbf{v}))$ (рис. 1.1, а). Для вогнутой функции хорда, соединяющая точки $(\mathbf{u}, f(\mathbf{u}))$ и $(\mathbf{v}, f(\mathbf{v}))$, проходит не выше графика функции между точками \mathbf{u} и \mathbf{v} (рис. 1.1, б).

Теорема 1.1. *Если функция $f(\mathbf{x})$ является выпуклой, то она в любой точке локального минимума достигает своего наименьшего значения.*

Доказательство. Пусть \mathbf{x}^* — точка локального минимума функции $f(\mathbf{x})$. Предположим, что в этой точке значение функции $f^* = f(\mathbf{x}^*)$ не является наименьшим, то есть существует точка \mathbf{y}^* , для которой $f(\mathbf{y}^*) < f(\mathbf{x}^*)$. Рассмотрим сечение функции $f(\mathbf{x})$, проходящее через пару точек \mathbf{y}^* , \mathbf{x}^* и определяющее функцию одной переменной $\varphi(\lambda) = f(\lambda \mathbf{y}^* + (1-\lambda)\mathbf{x}^*)$. Тогда с учетом неравенства (1.2) при $\lambda \in (0; 1)$

$$\varphi(\lambda) = f(\lambda \mathbf{y}^* + (1-\lambda)\mathbf{x}^*) \leq \lambda f(\mathbf{y}^*) + (1-\lambda)f(\mathbf{x}^*).$$

Отсюда, поскольку $f(\mathbf{y}^*) < f(\mathbf{x}^*)$, имеем

$$\varphi(\lambda) < \lambda f(\mathbf{x}^*) + (1-\lambda)f(\mathbf{x}^*) = f(\mathbf{x}^*) = \varphi(0).$$

Это означает, что в точке $\lambda = 0$ функция $\varphi(\lambda)$ достигает на отрезке $[0; 1]$ своего наибольшего значения. Для произвольной окрестности точки \mathbf{x}^* существует такое малое число $\lambda > 0$, что $\mathbf{x} = \lambda \mathbf{y}^* + (1-\lambda)\mathbf{x}^*$ принадлежит этой окрестности, то есть $\|\mathbf{x} - \mathbf{x}^*\| < \lambda$. Тогда $f(\mathbf{x}) = \varphi(\lambda) < \varphi(0) = f(\mathbf{x}^*)$. А это противоречит тому, что точка \mathbf{x}^* есть точка локального минимума. \square

Из доказанной теоремы вытекают важные следствия.

Следствие 1.1. *Во всех точках локального минимума выпуклая функция $f(\mathbf{x})$ имеет одно и то же значение, равное наименьшему значению f^* этой функции, а локальный минимум является и глобальным минимумом.*

Следствие 1.2. *Строго выпуклая функция может иметь не более одной точки локального минимума.*

Доказательство. Предположим, что \mathbf{x}^* и \mathbf{y}^* – две различные точки локального минимума строго выпуклой функции $f(\mathbf{x})$ и в этих точках функция достигает своего наименьшего значения f^* . Поскольку функция $f(\mathbf{x})$ строго выпуклая, то неравенство (1.2) является строгим. Тогда для любого $\lambda \in (0; 1)$ и $\mathbf{x} = \lambda \mathbf{x}^* + (1 - \lambda) \mathbf{y}^*$ имеем:

$$f(\mathbf{x}) = f(\lambda \mathbf{x}^* + (1 - \lambda) \mathbf{y}^*) < \lambda f(\mathbf{x}^*) + (1 - \lambda) f(\mathbf{y}^*) = \lambda f^* + (1 - \lambda) f^* = f^*.$$

Мы получили, что $f(\mathbf{x}) < f^*$, а это невозможно, так как f^* – наименьшее значение функции. \square

Численные методы решения задачи (1.1) называются *методами безусловной минимизации* или *методами минимизации без ограничений*. Существуют следующие классы методов безусловной минимизации.

1. Методы нулевого порядка (прямого поиска), не использующие производные целевой функции $f(\mathbf{x})$.

2. Методы первого порядка, использующие первые частные производные целевой функции $f(\mathbf{x})$.

3. Методы второго порядка, использующие матрицу вторых частных производных целевой функции $f(\mathbf{x})$.

Рассматриваемые в данном пособии методы оптимизации предназначены для минимизации унимодальных функций, а также для вычисления локального минимума целевой функции.

Простейшей функцией $f(\mathbf{x})$ нескольких переменных является

функция двумерного вектора переменных $\mathbf{x} = (x_1, x_1)^T$, $n = 2$. *Линией уровня* функции двух переменных называется линия, во всех точках которой функция принимает постоянное значение. Уравнение линии уровня имеет вид $f(\mathbf{x}) = C$, где C – некоторая постоянная величина. Изменяя значение C , получим семейство линий уровня.

Для функции трех переменных с $n = 3$ уравнение $f(\mathbf{x}) = C$ представляет *поверхность уровня*. Для $n > 3$ уравнение $f(\mathbf{x}) = C$ представляет *гиперповерхность уровня*.

Пример 1.1. Рассмотрим квадратичную функцию

$$f(\mathbf{x}) = x_1^2 + x_2^2 - 1,2x_1x_2. \quad (1.3)$$

Преобразуем эту функцию, выделяя полный квадрат:

$$f(\mathbf{x}) = (x_1^2 - 1,2x_1x_2 + 0,36x_2^2) - 0,36x_2^2 + x_2^2 = (x_1 - 0,6x_2)^2 + 0,64x_2^2.$$

Это неотрицательная функция и для нее $f(\mathbf{x}) \geq 0$. Она является суммой квадратов функций $f_1(\mathbf{x}) = x_1 - 0,6x_2$ и $f_2(\mathbf{x}) = 0,8x_2$. Приравняв нулю эти функции, получим систему уравнений: $x_1 = 0,6x_2$, $x_2 = 0$. Система уравнений имеет единственное решение $\mathbf{x}^* = (0; 0)^T$. Это точка строгого глобального минимума функции (1.3) с минимальным ее значением $f^* = 0$, поскольку для всех $\mathbf{x} \neq \mathbf{x}^*$ будет $f(\mathbf{x}) > 0$. График функции (1.3) представлен на рис. 1.2, где точка минимума отмечена звездочкой. На рис. 1.3 показаны линии уровня этой функции, которые представляют собой эллипсы с соответствующими значениями функции. Функция (1.3) соответствует эллиптическому параболоиду.

Пример 1.2. Рассмотрим функцию Розенброка

$$f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2. \quad (1.4)$$

Это также неотрицательная функция. Она представляет собой сумму квадратов функций $f_1(\mathbf{x}) = 10(x_2 - x_1^2)$ и $f_2(\mathbf{x}) = 1 - x_1$, приравнявая

нулю которые имеем систему уравнений: $x_2 = x_1^2$, $x_1 = 1$. Полученная система имеет единственное решение $\mathbf{x}^* = (1; 1)^T$. Это точка строгого глобального минимума функции Розенброка с минимальным значением функции $f^* = 0$, так как для всех $\mathbf{x} \neq \mathbf{x}^*$ будет $f(\mathbf{x}) > 0$.

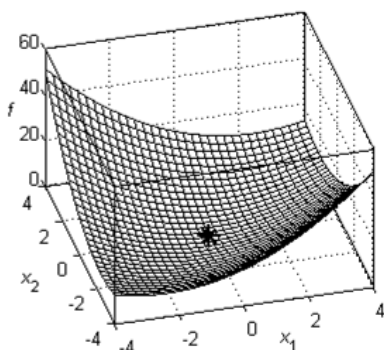


Рис. 1.2. График функции

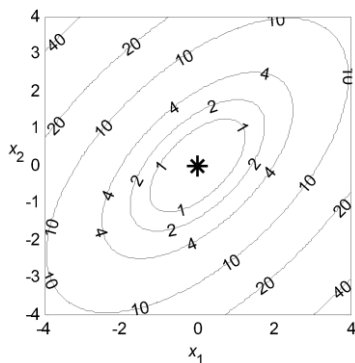


Рис. 1.3. Линии уровня функции

График функции (1.4) представлен на рис. 1.4. На рис. 1.5 показаны линии уровня этой функции, которые вытянуты вдоль параболической линии $x_2 = x_1^2$, определяющей дно искривленного оврага. Поскольку линии уровня напоминают линии разреза банана, функцию Розенброка называют также «бананообразной» функцией. Она является одной из основных тестовых функций для проверки эффективности многомерных методов безусловной минимизации, которые стартуют из начальной точки $\mathbf{x}_0 = (-1.2; 1)^T$. На рис. 1.4 и 1.5 отмечены: звездочкой – точка минимума \mathbf{x}^* , кругом – начальная точка \mathbf{x}_0 , штриховой линией – дно оврага функции Розенброка. Поскольку начальная точка расположена на дальнем склоне оврага относительно точки минимума функции, то для достижения точки минимума метод оптимизации должен преодолеть искривленный овраг с одним поворотом, что является непростой задачей.

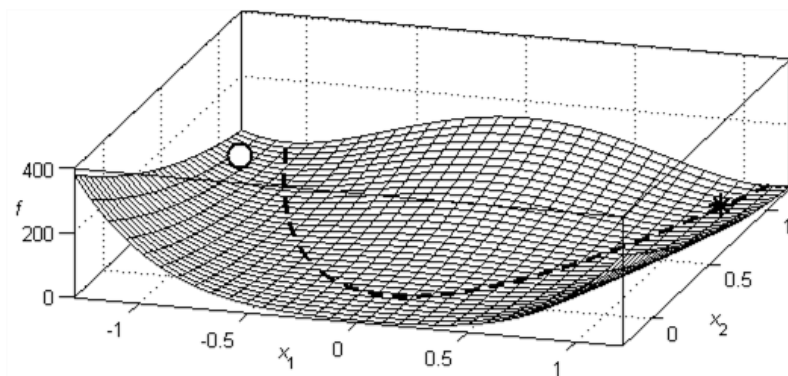


Рис. 1.4. График функции Розенброка

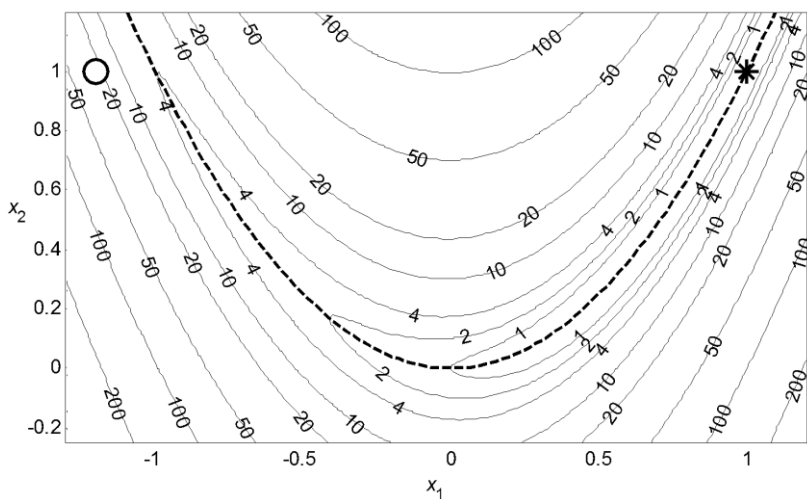


Рис. 1.5. Линии уровня функции Розенброка

Только тот численный метод оптимизации, который сможет минимизировать функцию Розенброка из заданной начальной точки, может считаться эффективным для минимизации гладких функций.

1.2. Условия оптимальности первого порядка

При изучении задач оптимизации важное место занимает вопрос об *условиях оптимальности*. *Необходимые* условия оптимальности – это условия, которым должна удовлетворять точка, являющаяся решением задачи. *Достаточные* условия оптимальности – это условия, из которых следует, что данная точка является решением задачи. Большое значение условий оптимальности определяется тем, что они составляют основу качественных методов изучения свойств экстремальных задач, используются при обосновании численных методов оптимизации, позволяют в простых случаях явно решить задачу.

Пусть целевая функция $f(\mathbf{x})$ является дифференцируемой при всех $\mathbf{x} \in R^n$. Вектор-столбец частных производных функции $f(\mathbf{x})$ в точке \mathbf{x} называется *градиентом* и обозначается

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial x_1}, \frac{\partial f(\mathbf{x})}{\partial x_2}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_n} \right)^T. \quad (1.5)$$

В формулах градиент часто обозначают вектором $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$.

Необходимое условие существования экстремума функции нескольких переменных $f(\mathbf{x})$ в точке \mathbf{x}^* представляется теоремой, аналогичной теореме для экстремума функции одной переменной.

Теорема 1.2. *Если дифференцируемая функция $f(\mathbf{x})$ имеет в точке \mathbf{x}^* экстремум, то*

$$\nabla f(\mathbf{x}^*) = \mathbf{0}. \quad (1.6)$$

Доказательство. Пусть для определенности функция $f(\mathbf{x})$ имеет в точке \mathbf{x}^* минимум. Разложим функцию $f(\mathbf{x})$ в окрестности точки \mathbf{x}^* в ряд Тейлора, ограничиваясь слагаемыми не выше первого порядка малости,

$$f(\mathbf{x}^* + \Delta \mathbf{x}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T \Delta \mathbf{x} + o(\|\Delta \mathbf{x}\|).$$

Положим $\Delta \mathbf{x} = -\alpha \nabla f(\mathbf{x}^*)$, где $\alpha > 0$ – некоторое малое число. Тогда с точностью до бесконечно малых первого порядка имеем

$$0 \leq f(\mathbf{x}^* + \Delta \mathbf{x}) - f(\mathbf{x}^*) = -\alpha \nabla f(\mathbf{x}^*)^T \nabla f(\mathbf{x}^*).$$

Разделив это выражение на α , придем к неравенству

$$0 \leq -\|\nabla f(\mathbf{x}^*)\|^2,$$

откуда следует, что $\nabla f(\mathbf{x}^*) = \mathbf{0}$. \square

Итак, для дифференцируемой по всем n переменным функции по этой теореме с учетом (1.5) и (1.6) получим:

$$\frac{\partial f(\mathbf{x}^*)}{\partial x_i} = 0, \quad i = \overline{1, n}. \quad (1.7)$$

Точка \mathbf{x}^* , удовлетворяющая этому условию, называется *стационарной* точкой. Это необходимое условие экстремума, общее для минимума и максимума. Данному условию также удовлетворяют *седловые* точки, которые соответствуют минимуму функции по одним направлениям и ее максимуму по другим.

Локальный минимум функции может достигаться и в точках, в которых первые частные производные функции не существуют. Такие точки называются *критическими*. Точку глобального минимума функции, если она существует, можно искать, сравнивая значения функции во всех стационарных и критических точках. Выяснять, какие из этих точек являются точками локального минимума, не обязательно.

Использование необходимого условия локального экстремума приводит к решению системы нелинейных уравнений (1.7), что является сложной задачей. Функция может иметь большое или даже бесконечное число стационарных точек, и тогда выбрать среди них точку с наименьшим значением функции сложно. Для выпуклых функций эта

задача существенно упрощается.

Теорема 1.3. Пусть функция $f(\mathbf{x})$ выпукла и дифференцируема в точке \mathbf{x}^* . Тогда, если $\nabla f(\mathbf{x}^*) = \mathbf{0}$, то \mathbf{x}^* – точка глобального минимума функции $f(\mathbf{x})$.

Доказательство. Возьмем произвольную точку $\mathbf{x} \in R^n$ и рассмотрим сечение функции $f(\mathbf{x})$, проходящее через точки \mathbf{x} , \mathbf{x}^* и определяющее функцию одной переменной $\varphi(\lambda) = f(\lambda\mathbf{x} + (1-\lambda)\mathbf{x}^*)$, где $\lambda \in [0; 1]$. Для такой функции $\varphi(0) = f(\mathbf{x}^*)$, $\varphi(1) = f(\mathbf{x})$. Из выпуклости функции $f(\mathbf{x})$ следует и выпуклость ее сечения $\varphi(\lambda)$, поэтому с учетом свойства (1.2) выпуклой функции $\varphi(\lambda) \leq \lambda\varphi(1) + (1-\lambda)\varphi(0)$. Отсюда $\varphi(\lambda) - \varphi(0) \leq \lambda[\varphi(1) - \varphi(0)]$, $[\varphi(\lambda) - \varphi(0)]/\lambda \leq \varphi(1) - \varphi(0)$ и

$$\varphi(1) - \varphi(0) \geq \lim_{\lambda \rightarrow 0} \frac{\varphi(\lambda) - \varphi(0)}{\lambda}.$$

Тогда $\varphi(1) - \varphi(0) \geq \varphi'(0)$. По правилу дифференцирования сложной функции $\varphi'(0) = \nabla f(\mathbf{x}^*)^T (\mathbf{x} - \mathbf{x}^*)$, и в силу условия $\nabla f(\mathbf{x}^*) = \mathbf{0}$ имеем $\varphi(1) - \varphi(0) \geq 0$, то есть $f(\mathbf{x}) \geq f(\mathbf{x}^*)$. \square

Доказанная теорема является достаточным условием минимума выпуклой функции. Из этой теоремы и теоремы 1.2 вытекает необходимое и достаточное условие минимума выпуклой функции.

Следствие 1.3. Пусть функция $f(\mathbf{x})$ выпукла и дифференцируема в точке \mathbf{x}^* . Для того, чтобы точка \mathbf{x}^* была точкой глобального минимума функции $f(\mathbf{x})$, необходимо и достаточно, чтобы выполнялось условие $\nabla f(\mathbf{x}^*) = \mathbf{0}$.

Необходимое условие оптимальности первого порядка (1.7) для выпуклых функций является и достаточным условием. В этом случае локальный минимум является также и глобальным.

1.3. Условия оптимальности второго порядка

Необходимое условие оптимальности первого порядка в общем случае не является достаточным условием оптимальности, поскольку стационарная точка не обязана быть решением задачи безусловной минимизации (1.1). Для анализа стационарных точек применяются условия оптимальности второго порядка, основанные на вторых производных целевой функции.

Пусть функция $f(\mathbf{x})$ является дважды дифференцируемой и существует матрица вторых частных производных целевой функции – матрица Гессе

$$\nabla^2 f(\mathbf{x}) = \mathbf{H}(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{pmatrix}. \quad (1.8)$$

В силу теоремы Шварца

$$\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\mathbf{x})}{\partial x_j \partial x_i}, \quad i, j = \overline{1, n}, \quad i \neq j.$$

Поэтому матрица Гессе (1.8) – симметрическая матрица и для нее

$$\mathbf{H}(\mathbf{x})^T = \mathbf{H}(\mathbf{x}). \quad (1.9)$$

Для строго выпуклой функции матрица Гессе является положительно определенной матрицей, а для строго вогнутой функции матрица Гессе отрицательно определенная.

Для выявления лишних стационарных точек может использоваться необходимое условие оптимальности второго порядка.

Теорема 1.4. Пусть функция $f(\mathbf{x})$ дважды дифференцируема в точке \mathbf{x}^* . Тогда, если \mathbf{x}^* — точка локального минимума функции $f(\mathbf{x})$, то матрица Гессе этой функции в точке \mathbf{x}^* $\nabla^2 f(\mathbf{x}^*)$ неотрицательно определена:

$$\mathbf{d}^T \nabla^2 f(\mathbf{x}^*) \mathbf{d} \geq 0, \quad \forall \mathbf{d} \in R^n. \quad (1.10)$$

Доказательство. Разложим функцию $f(\mathbf{x})$ в окрестности точки \mathbf{x}^* в ряд Тейлора, ограничиваясь слагаемыми не выше второго порядка малости,

$$f(\mathbf{x}^* + \lambda \mathbf{d}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T (\lambda \mathbf{d}) + \frac{1}{2!} (\lambda \mathbf{d})^T \nabla^2 f(\mathbf{x}^*) (\lambda \mathbf{d}) + o(\lambda^2).$$

Тогда с учетом (1.6) при достаточно малых λ имеем

$$0 \leq f(\mathbf{x}^* + \lambda \mathbf{d}) - f(\mathbf{x}^*) = \frac{1}{2!} \lambda^2 \mathbf{d}^T \nabla^2 f(\mathbf{x}^*) \mathbf{d} + o(\lambda^2).$$

Разделив обе части неравенства на $\lambda^2/2$ и переходя к пределу при $\lambda \rightarrow 0$, получим (1.10). \square

Достаточное условие локальной оптимальности содержит характерное усиление требований к матрице Гессе.

Теорема 1.5. Пусть функция $f(\mathbf{x})$ дважды дифференцируема в точке \mathbf{x}^* . Тогда, если $\nabla f(\mathbf{x}^*) = \mathbf{0}$ и матрица Гессе функции $f(\mathbf{x})$ в точке \mathbf{x}^* положительно определена, то есть

$$\mathbf{d}^T \nabla^2 f(\mathbf{x}^*) \mathbf{d} > 0, \quad \forall \mathbf{d} \in R^n, \quad \mathbf{d} \neq \mathbf{0}, \quad (1.11)$$

то \mathbf{x}^* — точка строгого локального минимума функции $f(\mathbf{x})$.

Доказательство. Доказательство проведем от противного. Предположим, что существуют точки, близкие к точке \mathbf{x}^* , со значением функции, не большим, чем в точке \mathbf{x}^* . Составим из них последова-

тельность $\{\mathbf{x}_k\}$, удовлетворяющую условиям:

$$\mathbf{x}_k \neq \mathbf{x}^*, \quad \mathbf{x}_k \rightarrow \mathbf{x}^*, \quad f(\mathbf{x}_k) \leq f(\mathbf{x}^*).$$

Представим \mathbf{x}_k в виде $\mathbf{x}_k = \mathbf{x}^* + \lambda_k \mathbf{d}_k$, где $\lambda_k = \|\mathbf{x}_k - \mathbf{x}^*\|$, $\mathbf{d}_k = (\mathbf{x}_k - \mathbf{x}^*)/\lambda_k$. Поскольку $\|\mathbf{d}_k\| = 1$, то без ограничения общности можно считать, что $\mathbf{d}_k \rightarrow \mathbf{d} \neq \mathbf{0}$. Составим ряд Тейлора, ограничиваясь слагаемыми не выше второго порядка малости

$$f(\mathbf{x}_k) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T (\lambda_k \mathbf{d}_k) + \frac{1}{2!} (\lambda_k \mathbf{d}_k)^T \nabla^2 f(\mathbf{x}^*) (\lambda_k \mathbf{d}_k) + o(\lambda_k^2).$$

Тогда с учетом (1.6) имеем

$$0 \geq f(\mathbf{x}_k) - f(\mathbf{x}^*) = \frac{1}{2} \lambda_k^2 \mathbf{d}_k^T \nabla^2 f(\mathbf{x}^*) \mathbf{d}_k + o(\lambda_k^2).$$

Разделив обе части неравенства на $\lambda_k^2/2$ и переходя к пределу при $k \rightarrow \infty$, получим противоречие с (1.11). \square

Для функции $f(x)$ скалярного аргумента x условия (1.10) и (1.11) означают, что вторая производная $f''(x^*)$ неотрицательна и положительна соответственно.

Заметим, что необходимые условия оптимальности, представленные теоремами 1.2 и 1.4, не являются достаточными. Например, для функции $f(x) = x^3$ эти условия выполнены в точке перегиба $\bar{x} = 0$, которая не является точкой минимума. Условия теоремы 1.5 не являются необходимыми для оптимальности. Для функции $f(x) = x^4$ эти условия не выполнены в точке строгого минимума $x^* = 0$, поскольку $f'(x^*) = 0$ и $f''(x^*) = 0$.

В тех случаях, когда функция $f(\mathbf{x})$ достаточно проста, теоремы 1.2, 1.3 и 1.4 позволяют явно решить задачу (1.1). При этом для

исследования матрицы $\nabla^2 f(\mathbf{x}^*)$ на положительную и отрицательную определенность, как правило, используется критерий Сильвестра – Якоби.

Теорема 1.6. Пусть \mathbf{x}^* является стационарной точкой функции $f(\mathbf{x})$, для которой $\nabla f(\mathbf{x}^*) = 0$, и пусть в этой точке и в некоторой ее окрестности функция $f(\mathbf{x})$ имеет непрерывные частные производные первого и второго порядков. Тогда:

1) если определитель матрицы Гессе

$$\Delta_n(\mathbf{x}^*) = \begin{vmatrix} \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_2 \partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x}^*)}{\partial x_n^2} \end{vmatrix}$$

и все его главные диагональные миноры $\Delta_1(\mathbf{x}^*)$, $\Delta_2(\mathbf{x}^*)$, ..., $\Delta_{n-1}(\mathbf{x}^*)$ положительны, то в точке \mathbf{x}^* функция $f(\mathbf{x})$ имеет минимум;

2) если определители $\Delta_1(\mathbf{x}^*)$, $\Delta_2(\mathbf{x}^*)$, ..., $\Delta_{n-1}(\mathbf{x}^*)$, $\Delta_n(\mathbf{x}^*)$ имеют соответственно знаки $-$, $+$, $-$, $+$, ..., то в точке \mathbf{x}^* функция $f(\mathbf{x})$ имеет максимум;

3) если $\Delta_n(\mathbf{x}^*) \neq 0$, а числа $\Delta_1(\mathbf{x}^*)$, $\Delta_2(\mathbf{x}^*)$, ..., $\Delta_{n-1}(\mathbf{x}^*)$, $\Delta_n(\mathbf{x}^*)$ ни положительны, ни знакопереваются по закону $-$, $+$, $-$, $+$, ..., то в точке \mathbf{x}^* функция $f(\mathbf{x})$ не имеет экстремума.

Примечание 1.1. Теорема 1.6 решает задачу исследования функции нескольких переменных на экстремум не полностью. Возможен случай, когда $\Delta_n(\mathbf{x}^*) = 0$. В этом случае полный дифференциал второго порядка $d^2 f(\mathbf{x}^*) = 0$, и тогда разность $f(\mathbf{x}) - f(\mathbf{x}^*)$ эквива-

лентна полному дифференциалу функции $f(\mathbf{x})$ третьего порядка, исследование которого, вообще говоря, намного сложнее. В этом случае вопрос о наличии экстремума в точке \mathbf{x}^* остается открытым и требует дополнительного исследования.

Пр и м е р 1.3. Для квадратичной функции (1.3) имеем:

$$\nabla f(\mathbf{x}) = \begin{pmatrix} 2x_1 - 1,2x_2 \\ 2x_2 - 1,2x_1 \end{pmatrix}, \quad \nabla^2 f(\mathbf{x}) = \begin{pmatrix} 2 & -1,2 \\ -1,2 & 2 \end{pmatrix}.$$

Необходимое условие оптимальности первого порядка $\nabla f(\mathbf{x}) = \mathbf{0}$ приводит к системе уравнений: $2x_1 - 1,2x_2 = 0$, $2x_2 - 1,2x_1 = 0$. Решая эту систему, получим стационарную точку $\mathbf{x}^* = (0; 0)^T$. Определитель матрицы Гессе $\nabla^2 f(\mathbf{x})$ и его главный диагональный минор положительны:

$$\Delta_1(\mathbf{x}^*) = 2 > 0, \quad \Delta_2(\mathbf{x}^*) = \begin{vmatrix} 2 & -1,2 \\ -1,2 & 2 \end{vmatrix} = 2^2 - 1,2^2 = 2,56 > 0.$$

По теореме 1.6 найденная точка \mathbf{x}^* – точка минимума функции (1.3), что совпадает с ранее полученным результатом в примере 1.1.

Пр и м е р 1.4. Для функции Розенброка (1.4) получим:

$$\nabla f(\mathbf{x}) = \begin{pmatrix} -400x_1(x_2 - x_1^2) - 2(1 - x_1) \\ 200(x_2 - x_1^2) \end{pmatrix},$$

$$\nabla^2 f(\mathbf{x}) = \begin{pmatrix} 1200x_1^2 - 400x_2 + 2 & -400x_1 \\ -400x_1 & 200 \end{pmatrix}.$$

Градиент и матрица Гессе функции Розенброка зависят от вектора \mathbf{x} . Необходимое условие экстремума $\nabla f(\mathbf{x}) = \mathbf{0}$ дает систему уравнений: $200x_1(x_2 - x_1^2) = x_1 - 1$, $x_2 = x_1^2$. Решением этой системы найдем стаци-

онарную точку $\mathbf{x}^* = (1; 1)^T$, в которой

$$\nabla^2 f(\mathbf{x}^*) = \begin{pmatrix} 802 & -400 \\ -400 & 200 \end{pmatrix}.$$

Определитель матрицы Гессе и его главный минор положительны:

$$\Delta_1(\mathbf{x}^*) = 802 > 0, \quad \Delta_2(\mathbf{x}^*) = 802 \cdot 200 - 400^2 = 400 > 0.$$

Следовательно, \mathbf{x}^* – точка минимума функции (1.4), ранее найденная в примере 1.2.

1.4. Метод циклического покоординатного спуска

Этот метод является простейшим численным методом безусловной оптимизации целевой функции n переменных $f(\mathbf{x})$ и основан на одномерной минимизации функции последовательно по каждой из n переменных. Метод заключается в последовательном проведении одномерного поиска в направлениях каждой из n координатных осей. Для этого задаются единичные векторы – орты координатных осей:

$$\mathbf{e}_1 = (1, 0, 0, \dots, 0)^T, \quad \mathbf{e}_2 = (0, 1, 0, \dots, 0)^T, \quad \dots, \quad \mathbf{e}_n = (0, 0, 0, \dots, 1)^T.$$

Вначале проводится одномерный поиск из заданной начальной точки \mathbf{x}_0 в направлении орта \mathbf{e}_1 , и вычисляется точка минимума функции скалярного аргумента $\varphi(\lambda) = f(\mathbf{x}_0 + \lambda \mathbf{e}_1)$

$$\lambda_1 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{e}_1).$$

Затем определяется следующая точка поиска

$$\mathbf{x}_1 = \mathbf{x}_0 + \lambda_1 \mathbf{e}_1.$$

Из точки \mathbf{x}_1 проводится одномерный поиск в направлении орта \mathbf{e}_2 :

$$\lambda_2 = \arg \min_{\lambda} f(\mathbf{x}_1 + \lambda \mathbf{e}_2), \quad \mathbf{x}_2 = \mathbf{x}_1 + \lambda_2 \mathbf{e}_2.$$

Аналогично проводится одномерный поиск в направлениях всех остальных ортов. После n таких одномерных поисков получаем точку

$$\mathbf{x}_n = \mathbf{x}_{n-1} + \lambda_n \mathbf{e}_n.$$

На этом первый цикл метода заканчивается, и переходят к следующему циклу. При этом из точки \mathbf{x}_n опять проводится одномерный поиск в направлении орта \mathbf{e}_1 и так далее. Вычисления производятся до тех пор, пока расстояние между лучшими точками последовательных циклов поиска не превысит заданной допустимой погрешности ε .

Алгоритм метода покоординатного спуска.

Входные параметры: \mathbf{x} – начальная точка поиска; $f(\mathbf{x})$ – процедура вычисления функции; ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Положить $n = \dim \mathbf{x}$.
2. Положить $\mathbf{y} = \mathbf{x}$, $i = 1$.
3. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{e}_i)$.
4. Положить $\mathbf{x} = \mathbf{x} + r \cdot \mathbf{e}_i$.
5. Если $i < n$, то положить $i = i + 1$ и перейти к шагу 3.
6. Если $\|\mathbf{x} - \mathbf{y}\| > \varepsilon$, то перейти к шагу 2.
7. Остановиться.

В этом алгоритме шаги 3–5 производят последовательный одномерный поиск в направлениях ортов осей координат. Шаги 2–6 составляют итерационный цикл метода. Одномерный поиск обычно проводится комбинированным алгоритмом, включающим алгоритм метода Свенна для нахождения интервала неопределенности, содержащего минимум, и один из алгоритмов уменьшения этого интервала. Уменьшение интервала проводится до тех пор, пока точность вычисления параметра λ не превысит заданного малого числа ε_1 . Приведенный

алгоритм прост в реализации, но эффективен лишь в случаях, когда целевая функция является *сепарабельной*, то есть представляет собой сумму функций, каждая из которых зависит лишь от одной переменной.

Пример 1.5. На рис. 1.6 представлена траектория поиска минимума квадратичной функции (1.3) методом циклического покоординатного спуска, включающая все точки поиска. Начальная точка $\mathbf{x}_0 = (4; 1)^T$ отмечена кругом, конечная точка указана ромбом. Лучшие точки итераций соединены жирной линией. Для нахождения точки минимума $\mathbf{x}^* = (0; 0)^T$ с допустимой погрешностью 10^{-3} метод затратил 9 итераций с двумя одномерными поисками на каждой итерации. Значения целевой функции были вычислены 76 раз.

Здесь и в последующих вычислительных примерах одномерный поиск при начальном единичном шаге $h_0 = 1$ и допустимой погрешности $\varepsilon_1 = 10^{-5}$ производится комбинацией метода Свенна и метода квадратичной интерполяции с тремя точками. Погрешность одномерного поиска задана на два порядка выше допустимой погрешности метода многомерной минимизации $\varepsilon = 10^{-3}$. Повышенная точность одномерного поиска используется для сопоставления теоретических свойств методов, обоснованных при условии выполнения точного одномерного поиска, и результатов их численной реализации. Для уменьшения количества вычислений функции и во избежание преждевременного уменьшения шага начальный шаг одномерного поиска задавался как взвешенная сумма двух выполненных предыдущих шагов $h_{k+1} = 0,75s_k + 0,25s_{k-1}$, начиная со второй итерации при $s_0 = 1$.

Пример 1.6. На рис. 1.7 представлена траектория поиска минимума функции Розенброка (1.4) методом циклического покоординатного спуска, включающая лучшие точки итераций. Процесс минимизации занял 355 итераций и прекратился при уменьшении величины шага до 10^{-3} . Было произведено 3164 вычисления функции. Точка минимума, отмеченная звездочкой, не была найдена.

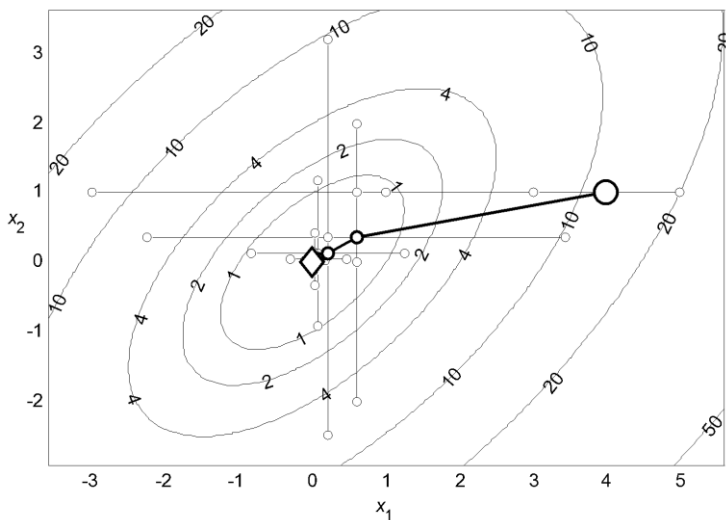


Рис. 1.6. Минимизация квадратичной функции методом циклического покоординатного спуска

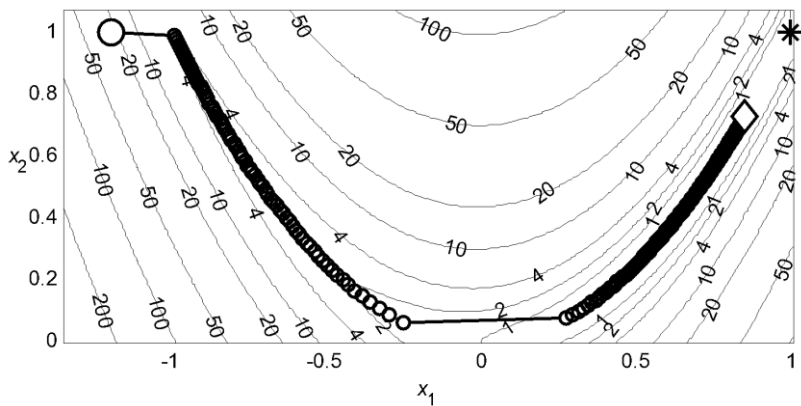


Рис. 1.7. Минимизация функции Розенброка методом покоординатного спуска

Достоинством метода циклического покоординатного спуска является его простота. Недостатком метода является его неэффектив-

ность при минимизации овражных функций. Однако это фундаментальный метод многомерной оптимизации, поскольку идея этого метода о последовательном применении одномерного поиска в n направлениях используется в других более эффективных методах оптимизации.

Метод циклического покоординатного спуска и его модификации также называют методом циклического покоординатного поиска, циклическим координатным методом, методом Гаусса-Зейделя, методом локальных вариаций.

1.5. Методы спуска

С помощью рассмотренного метода циклического покоординатного спуска выполняется одномерный поиск из заданной точки в направлении, параллельном одной из координатных осей, до точки минимума в данном направлении. Затем одномерный поиск производится в направлениях, параллельных другим осям. Все направления поиска в этом методе фиксированы. Целесообразно модифицировать этот метод таким образом, чтобы на каждом этапе одномерный поиск производился вдоль «наилучшего направления», обеспечивающего наиболее быстрое убывание функции. Важной концепцией методов многомерной минимизации является понятие о направлении спуска.

Вектор \mathbf{d} называется *направлением спуска* функции $f(\mathbf{x})$ в точке \mathbf{x} , если существует такое $\delta > 0$, что $f(\mathbf{x} + \lambda \mathbf{d}) < f(\mathbf{x})$ для всех $\lambda \in (0, \delta)$.

Пусть функция $f(\mathbf{x})$ дифференцируема и задан вектор $\mathbf{d} \in R^n$. Тогда производная функции в точке \mathbf{x} по направлению \mathbf{d} определяется как предел

$$f'(\mathbf{x}; \mathbf{d}) = \lim_{\lambda \rightarrow +0} \frac{f(\mathbf{x} + \lambda \mathbf{d}) - f(\mathbf{x})}{\lambda} = \nabla f(\mathbf{x})^T \mathbf{d}. \quad (1.12)$$

В частности, если $f'(\mathbf{x}; \mathbf{d}) < 0$, то \mathbf{d} является направлением спуска. Поэтому на основании (1.12) используют еще одно определение.

Пусть $f(\mathbf{x})$ – дифференцируемая функция при $\mathbf{x} \in R^n$. Если существует вектор $\mathbf{d} \in R^n$, такой что $\nabla f(\mathbf{x})^T \mathbf{d} < 0$, то \mathbf{d} называется *направлением спуска* функции $f(\mathbf{x})$ в точке \mathbf{x} .

Действительно, при помощи формулы Тейлора

$$f(\mathbf{x} + \lambda \mathbf{d}) = f(\mathbf{x}) + \lambda \nabla f(\mathbf{x})^T \mathbf{d} + o(\lambda)$$

легко видеть, что существует $\delta > 0$, такое, что $f(\mathbf{x} + \lambda \mathbf{d}) < f(\mathbf{x})$ для всех $\lambda \in (0, \delta)$, если и только если \mathbf{d} является направлением спуска функции $f(\mathbf{x})$ в точке \mathbf{x} , то есть $\nabla f(\mathbf{x})^T \mathbf{d} < 0$.

Методы многомерной минимизации функций, в которых используются направления спуска, называются *методами спуска*.

Методы спуска являются итерационными методами. Для выполнения первой итерации задается начальная точка \mathbf{x}_0 . Итерации выполняются по формуле

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad (1.13)$$

где k – номер итерации, $k = 0, 1, 2, \dots$; λ_k – скалярное значение, определяющее шаг поиска; \mathbf{d}_k – направление спуска. Итерации продолжают до тех пор, пока не выполнится критерий окончания процесса минимизации

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \leq \varepsilon, \quad (1.14)$$

где ε – допустимая погрешность, $\varepsilon > 0$.

Во всех методах спуска используется формула (1.13). Различные методы спуска отличаются выбором направления спуска \mathbf{d}_k и способом определения шага поиска λ_k .

Во многих методах шаг поиска λ_k вычисляется как такое значе-

ние переменной λ , которое доставляет минимум функции одной переменной $\varphi_k(\lambda) = f(\mathbf{x}_k + \lambda \mathbf{d}_k)$. Используя необходимое условие минимума дифференцируемой функции скалярного аргумента, получим, что в точке минимума λ_k должно выполняться равенство $\varphi'_k(\lambda_k) = 0$. Но это означает, что равна нулю производная (1.12) функции $f(\mathbf{x})$ в точке \mathbf{x}_{k+1} по направлению \mathbf{d}_k

$$\nabla f(\mathbf{x}_{k+1})^T \mathbf{d}_k = 0. \quad (1.15)$$

Это необходимое условие точного одномерного поиска. Таким образом, в методах спуска с применением точного одномерного поиска градиент функции в конечной точке итерации \mathbf{x}_{k+1} ортогонален направлению поиска \mathbf{d}_k в этой итерации (рис. 1.8).

Итерация метода спуска (1.13) с проведением точного одномерного поиска представляется в виде:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k). \quad (1.16)$$

Эти формулы будут использоваться в последующих численных методах многомерной безусловной минимизации, представленных в данном пособии и отличающихся лишь выбором направления спуска. Любой метод спуска можно описать общим алгоритмом, основанным на формулах (1.16), (1.14).

Алгоритм метода спуска.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить направление спуска \mathbf{d} в точке \mathbf{x} .

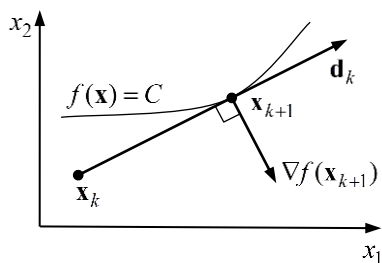


Рис. 1.8. Точный одномерный поиск

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.

4. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 1.

5. Остановиться.

В этом алгоритме на шаге 2 выполняется одномерный поиск минимума из текущей точки поиска в направлении спуска \mathbf{d} . Для повышения эффективности одномерного поиска производят масштабирование направления поиска и используют вектор направления $\mathbf{d}^0 = \mathbf{d} / \|\mathbf{d}\|$ единичной длины. Шаг перехода в следующую точку поиска обозначен через вектор \mathbf{s} . Итерации продолжаются, пока длина \mathbf{s} больше заданной допустимой погрешности.

1.6. Метод наискорейшего спуска

Для обоснования одного из простейших методов спуска представим дифференцируемую целевую функцию $f(\mathbf{x})$, где $\mathbf{x} \in R^n$, рядом Тейлора, ограничиваясь слагаемым первого порядка малости,

$$f(\mathbf{x} + \Delta \mathbf{x}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \Delta \mathbf{x} + o(\|\Delta \mathbf{x}\|). \quad (1.17)$$

Пусть \mathbf{x} – фиксированная начальная точка поиска, $\Delta \mathbf{x}$ – приращение аргумента, которое обеспечивает убывание функции, причем длина этого приращения $|\Delta \mathbf{x}|$ постоянна. При достаточно малых значениях $\Delta \mathbf{x}$, пренебрегая в (1.17) слагаемыми высшего порядка малости, имеем изменение функции

$$\Delta f = f(\mathbf{x} + \Delta \mathbf{x}) - f(\mathbf{x}) = \nabla f(\mathbf{x})^T \Delta \mathbf{x}. \quad (1.18)$$

В первом приближении изменение функции (1.18) равно скалярному произведению векторов $\nabla f(\mathbf{x})$ и $\Delta \mathbf{x}$. Пусть θ – угол между этими векторами. Тогда

$$\Delta f = |\nabla f(\mathbf{x})| \cdot |\Delta \mathbf{x}| \cdot \cos \theta.$$

При постоянной длине векторов $\nabla f(\mathbf{x})$ и $\Delta \mathbf{x}$ убывание функции будет наибольшим, если $\cos \theta = -1$ и $\theta = \pi$. Это означает, что векторы $\nabla f(\mathbf{x})$ и $\Delta \mathbf{x}$ противоположно направлены и $\Delta \mathbf{x} = -\lambda \nabla f(\mathbf{x})$, где $\lambda > 0$. Таким образом, направление наиболее быстрого убывания функции $f(\mathbf{x})$ в точке \mathbf{x} совпадает с *антиградиентом* $-\nabla f(\mathbf{x})$.

Направление антиградиента $-\nabla f(\mathbf{x})$ наиболее быстрого убывания функции $f(\mathbf{x})$ в точке \mathbf{x} называется *направлением наискорейшего спуска*. Градиент же функции $\nabla f(\mathbf{x})$ определяет направление наиболее быстрого возрастания функции $f(\mathbf{x})$ в точке \mathbf{x} .

Пусть $\Delta \mathbf{x}$ – такой малый шаг вдоль линии уровня $f(\mathbf{x}) = C$, что $f(\mathbf{x} + \Delta \mathbf{x}) = C$. Тогда $\Delta f = f(\mathbf{x} + \Delta \mathbf{x}) - f(\mathbf{x}) = 0$, и по формуле (1.18) с точностью до бесконечно малых первого порядка $\nabla f(\mathbf{x})^T \Delta \mathbf{x} = 0$. То есть в любой точке \mathbf{x} направление градиента $\nabla f(\mathbf{x})$ перпендикулярно линии уровня, проходящей через эту точку, поскольку вдоль этой линии функция постоянна (рис. 1.8). Это замечание касается и антиградиента $-\nabla f(\mathbf{x})$.

Метод минимизации целевой функции $f(\mathbf{x})$, в котором направление поиска определяется антиградиентом $-\nabla f(\mathbf{x})$, называется *методом наискорейшего спуска*. Это означает, что если на некотором шаге процесса оптимизации получена точка \mathbf{x}_k , то поиск минимума функции осуществляется вдоль направления $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$. В данном методе итерации выполняются по формуле (1.13)

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \lambda_k \nabla f(\mathbf{x}_k), \quad (1.19)$$

где λ_k – значение переменной λ , которое доставляет минимум функции $\varphi(\lambda) = f(\mathbf{x}_k - \lambda \nabla f(\mathbf{x}_k))$. Обозначая $\mathbf{g}_k = -\nabla f(\mathbf{x}_k)$ и учитывая

(1.19), запишем формулы итерации метода наискорейшего спуска:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \lambda_k \mathbf{g}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k - \lambda \mathbf{g}_k). \quad (1.20)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность, $\varepsilon > 0$. По формулам (1.20) составим алгоритм метода наискорейшего спуска.

Алгоритм метода наискорейшего спуска.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$.
2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} - \lambda \cdot \mathbf{g})$, $\mathbf{s} = -r \cdot \mathbf{g}$.
3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.
4. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 1.
5. Остановиться.

В этом алгоритме на шаге 2 выполняется одномерный поиск минимума из текущей точки поиска в направлении антиградиента $-\mathbf{g}$. Для повышения эффективности численной процедуры одномерного поиска производят масштабирование направления поиска и используют вектор направления $\mathbf{d}^0 = -\mathbf{g}/\|\mathbf{g}\|$ единичной длины. Шаг перехода в следующую точку поиска обозначен через \mathbf{s} . Итерации продолжаются, пока длина \mathbf{s} больше заданной допустимой погрешности.

Пример 1.7. На рис. 1.9 представлена траектория минимизации квадратичной функции (1.3) методом наискорейшего спуска, состоящая из лучших точек итераций. Градиент вычислялся по формулам конечных разностей. В соответствии с замечаниями примера 1.5 о проведении одномерного поиска начальное значение шага одномерного поиска на каждой итерации вычисляется на основании двух выпол-

ненных предыдущих итераций, а вектор направления поиска масштабирован по формуле $\mathbf{d}^0 = -\mathbf{g}/\|\mathbf{g}\|$. Для нахождения точки минимума с погрешностью 10^{-3} затрачено 13 итераций и 80 вычислений функции.

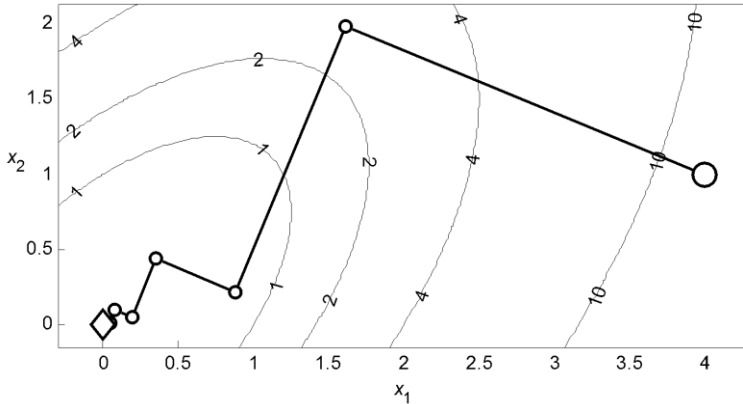


Рис. 1.9. Минимизация квадратичной функции методом наискорейшего спуска

При выполнении точного одномерного поиска из точки \mathbf{x}_k в направлении вектора \mathbf{d}_k по (1.15) получим $\mathbf{d}_k^T \mathbf{g}_{k+1} = 0$, то есть вектор направления ортогонален градиенту функции в точке минимума этой функции в данном направлении. Для метода наискорейшего спуска это означает, что $\mathbf{g}_k^T \mathbf{g}_{k+1} = 0$, то есть градиенты в последовательных точках поиска ортогональны. Следовательно, траектория метода состоит из последовательности ортогональных отрезков $\mathbf{s}_0, \mathbf{s}_1, \mathbf{s}_2, \dots$, где $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ — шаг перехода из точки \mathbf{x}_k в точку \mathbf{x}_{k+1} (рис. 1.9).

Пример 1.8. На рис. 1.10 представлена траектория поиска минимума функции Розенброка методом наискорейшего спуска. Процесс минимизации занял 306 итераций при 2060 вычислениях функции и прекратился при уменьшении величины шага многомерного поиска ниже значения 10^{-3} . Точка минимума не была найдена.

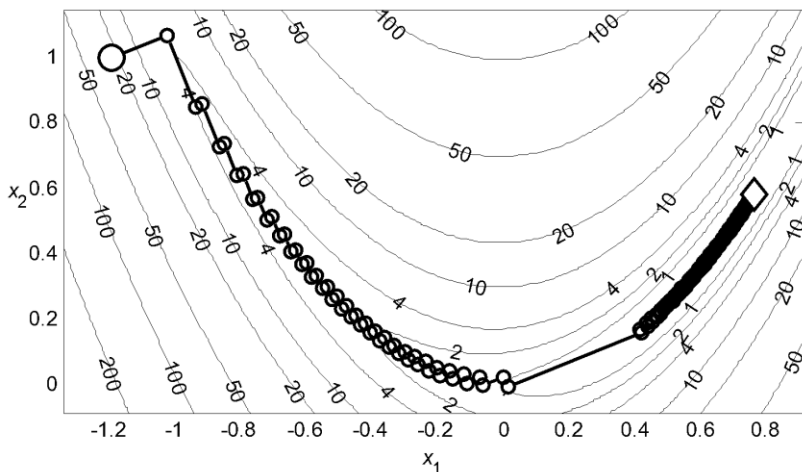


Рис. 1.10. Минимизация функции Розенброка методом наискорейшего спуска

Метод наискорейшего спуска называется также *методом Коши*, поскольку известный французский математик Огюстен Луи Коши первым в 1847 году представил этот метод для решения систем линейных уравнений.

Метод наискорейшего спуска является простейшим, наиболее известным и самым фундаментальным методом безусловной минимизации дифференцируемых функций нескольких переменных. Поскольку в нем используется отрицательный градиент как направление спуска, он также называется *градиентным методом*. Этот метод основан на первых производных целевой функции, поэтому он является методом первого порядка. Достоинством метода является его простота, но он обладает тем же основным недостатком, что и метод циклического покоординатного спуска, – низкой эффективностью при минимизации овражных функций (рис. 1.10).

Метод Коши, как правило, позволяет существенно уменьшить значение целевой функции при движении из начальных точек, расположенных на значительных расстояниях от точки минимума, и поэто-

му часто используется как начальная процедура в других методах минимизации. Разработаны различные модификации метода Коши. Однако существуют методы минимизации, которые основаны на других принципах и существенно превосходят метод Коши в эффективности.

1.7. Вычисление градиента

В методе наискорейшего спуска и в других методах оптимизации первого порядка для вычисления направления поиска используется градиент целевой функции $f(\mathbf{x})$. Во многих прикладных задачах оптимизации получение аналитического выражения для градиента затруднительно. Если значения целевой функции определяются с помощью некоторого алгоритма или в результате имитационного моделирования, то аналитическое выражение градиента вообще невозможно представить. Кроме того, аналитическое дифференцирование сложных функций трудоемко и не исключает возникновение ошибок. Целесообразно иметь возможность численного определения градиента целевой функции.

Простейшей формулой вычисления проекций градиента $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$ является конечная разность вперёд:

$$g_i(\mathbf{x}) = \frac{f(\mathbf{x} + \delta \mathbf{e}_i) - f(\mathbf{x})}{\delta}, \quad i = \overline{1, n}, \quad (1.21)$$

где δ – некоторое малое приращение, \mathbf{e}_i – орт i -той оси координат. Такая аппроксимация градиента непосредственно основана на определении частной производной функции многих переменных и при достаточно малых значениях δ может давать весьма точные оценки. Погрешность формулы (1.21) составляет $o(\delta)$, поэтому чем меньше δ , тем точнее вычислялись бы проекции антиградиента, если бы вычисления проводились с абсолютной точностью. В пределе при стремлении δ к нулю аппроксимация становится точной, однако это не может служить рекомендацией по выбору δ при машинных вычислениях.

Так как машинные вычисления обладают погрешностями представления вещественных чисел и округлений при вычислениях, то величина δ ограничивается снизу точностью машинных вычислений. Обычно эта точность определяется с помощью *машинного эпсилон* ε_m .

Машинное эпсилон ε_m – это наименьшее положительное число, для которого при машинном сложении чисел выполняется неравенство

$$1 + \varepsilon_m > 1.$$

Например, в компьютерной математической системе MATLAB ε_m задаётся с помощью встроенной константы $\text{eps} \approx 2 \cdot 10^{-16}$.

Выбор δ в формуле (1.21) должен осуществляться в зависимости от точности представления чисел в вычислительной системе, координат точки \mathbf{x} и вида функции $f(\mathbf{x})$. Величина δ должна выбираться достаточно большой, чтобы при погрешностях машинных вычислений в формуле (1.21) было $\mathbf{x} + \delta \mathbf{e}_i \neq \mathbf{0}$. Для решения большинства задач оптимизации достаточно положить в формуле (1.21) параметр $\delta = 10^{-8}$. По формуле разности вперед (1.21) составлен алгоритм вычисления градиента.

Алгоритм вычисления градиента.

Входные параметры: \mathbf{x} и $f_{\mathbf{x}}$ – точка, в которой вычисляется градиент, и значение в ней функции; $f(\mathbf{x})$ – процедура вычисления целевой функции; δ – параметр приращения аргумента.

Выходной параметр \mathbf{g} – значение градиента.

1. Положить $n = \dim \mathbf{x}$, $\mathbf{y} = \mathbf{x}$, $a = 1/\delta$, $i = 1$.
2. Положить $y_i = x_i + \delta$.
3. Вычислить $f_{\mathbf{y}} = f(\mathbf{y})$, $g_i = a \cdot (f_{\mathbf{y}} - f_{\mathbf{x}})$.
4. Положить $y_i = x_i$.
5. Если $i < n$, то положить $i = i + 1$ и перейти к шагу 2.
6. Остановиться.

По формуле разности вперёд (1.21) при известном значении функции $f(\mathbf{x})$ в точке \mathbf{x} для вычисления градиента требуется n дополнительных вычислений функции.

С помощью центральных разностей значения проекций градиента вычисляются по формуле:

$$g_i(\mathbf{x}) = \frac{f(\mathbf{x} + \delta \mathbf{e}_i) - f(\mathbf{x} - \delta \mathbf{e}_i)}{2\delta}, \quad i = \overline{1, n}. \quad (1.22)$$

Погрешность этой формулы составляет $o(\delta^2)$. Однако эта формула требует $2n$ дополнительных вычислений значений целевой функции. Поэтому в численных методах оптимизации первого порядка применяется формула конечной разности вперёд (1.21), требующая меньшего числа вычислений функции. Для примеров 1.7 и 1.8 с методом наискорейшего спуска использовалась формула (1.21). Формула же центральных разностей (1.22) применяется в методах второго порядка.

Лабораторная работа

Тема лабораторной работы.

Разработка подпрограмм для базовых методов многомерной безусловной минимизации.

Цель лабораторной работы.

Разработать подпрограмму для базового метода безусловной минимизации функций многих переменных.

Порядок выполнения лабораторной работы.

1. Получить у преподавателя вариант целевой функции $f(\mathbf{x})$ двух переменных с начальной точкой поиска \mathbf{x}_0 и задание на программирование одного из базовых методов безусловной минимизации:

- 1) метод циклического покоординатного спуска;
- 2) метод наискорейшего спуска.

2. Получить у преподавателя задание на программирование одно-

го из методов одномерного поиска в многомерном пространстве:

- 1) метод дихотомии;
- 2) метод деления интервала пополам;
- 3) метод Фибоначчи;
- 4) метод золотого сечения;
- 5) метод адаптации шага;
- 6) метод квадратичной интерполяции с тремя точками;
- 7) метод квадратичной интерполяции с двумя точками;
- 8) метод секущих;
- 9) метод касательных;
- 10) метод кубической интерполяции с четырьмя точками;
- 11) метод кубической интерполяции с двумя точками.

3. Аналитически найти точку \mathbf{x}^* минимума заданной функции $f(\mathbf{x})$ и вычислить минимальное значение функции $f^* = f(\mathbf{x}^*)$.

4. Написать подпрограмму вычисления значений функции $f(\mathbf{x})$ с входным параметром \mathbf{x} и выходным параметром f , а также подпрограмму вычисления градиента целевой функции.

5. Протестировать подпрограмму вычисления функции $f(\mathbf{x})$ при вычислении значений $f_0 = f(\mathbf{x}_0)$ и $f^* = f(\mathbf{x}^*)$, сравнить f^* с f_0 . Протестировать подпрограмму вычисления градиента целевой функции в точках \mathbf{x}_0 и \mathbf{x}^* .

6. На основании алгоритмов для заданного базового метода безусловной минимизации и метода одномерного поиска написать компьютерные подпрограммы в соответствии с требованиями:

1) входные параметры подпрограмм: \mathbf{x}_0 – начальная точка поиска, $f(\mathbf{x})$ – целевая функция, ε – допустимая погрешность;

2) выходные параметры подпрограмм: \mathbf{X} – массив всех точек поиска \mathbf{x}_k ; \mathbf{F} – массив соответствующих значений функции $f_k = f(\mathbf{x}_k)$; \mathbf{x} и $f_{\mathbf{x}}$ – лучшая точка поиска и значение в ней функции;

3) на каждой итерации выводить на экран компьютера строку, со-

державшую номер точки поиска k , шаг перехода в новую точку $h_k = \|\mathbf{s}_k\| = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|$, значение функции f_k и значение вектора переменных \mathbf{x}_k ;

4) итерации продолжаются до тех пор, пока выполняется неравенство $h_k > \varepsilon$;

5) после окончания процесса оптимизации на экране должна отображаться таблица, имеющая соответствующий заголовок и представляющая процесс минимизации функции;

6) под таблицей необходимо отобразить количество вычислений целевой функции, конечный шаг поиска, минимальное значение функции и соответствующую ему точку поиска.

7. Написать подпрограмму отображения процесса оптимизации на трехмерном графике функции $f(\mathbf{x})$ и на двумерном графике линий уровня этой функции в соответствии с требованиями:

1) входные параметры подпрограммы: $f(\mathbf{x})$, \mathbf{X} , \mathbf{F} ;

2) определить границы интервалов горизонтальных осей:

$$a_1 = \min\{X_{1k}\}, \quad a_2 = \min\{X_{2k}\}, \quad b_1 = \max\{X_{1k}\}, \quad b_2 = \max\{X_{2k}\};$$

3) отобразить на экране график функции $f(\mathbf{x})$ и график ее линий уровня в прямоугольнике $\mathbf{a} \leq \mathbf{x} \leq \mathbf{b}$, где $\mathbf{a} = (a_1, a_2)^T$, $\mathbf{b} = (b_1, b_2)^T$;

4) на построенных графиках отобразить все последовательные точки поиска (\mathbf{x}_k, f_k) , $k = 0, 1, 2, \dots$, соединенные отрезками прямой линии;

5) на тех же графиках отобразить начальную и конечную точки поиска маркерами, различными по форме и цвету.

8. Написать общую программу для выполнения вычислительного процесса минимизации целевой функции многих переменных и отображения на экране этого процесса в соответствии с требованиями:

1) задать в программе значения \mathbf{x}_0 , h , ε ;

2) путем вызова разработанной подпрограммы заданного базового

метода безусловной минимизации с использованием подпрограмм вычисления функции $f(\mathbf{x})$ и заданного метода одномерного поиска, а также значений \mathbf{x}_0 , h , ε выполнить итерационный процесс минимизации функции с построением таблицы процесса на экране и получением массивов точек поиска \mathbf{X} и значений функции \mathbf{F} , лучшей точки поиска \mathbf{x} и значения в ней функции $f_{\mathbf{x}}$;

3) с помощью подпрограммы графического отображения итерационного процесса по массивам \mathbf{X} и \mathbf{F} отобразить графики процесса минимизации целевой функции многих переменных на экране компьютера.

9. Выполнить анализ полученных результатов:

1) на основании табличной и графической информации о работе метода одномерного поиска в многомерном пространстве определить конечные точки метода Свенна и показать найденный этим методом интервал неопределенности;

2) на нескольких начальных итерациях по табличным данным и графикам сопоставить начало работы базового метода безусловной минимизации с его рабочими формулами;

3) оценить эффективность базового метода безусловной минимизации.

10. Ответить на вопросы преподавателя по теме лабораторной работы.

11. Оформить и сдать отчет о проведенной лабораторной работе.

Содержание отчета:

1) титульный лист установленного образца с указанием организации, названия учебной дисциплины, темы работы, номера варианта, исполнителя и принимающего, города, года;

2) постановка задачи минимизации целевой функции, включающая задание функции, начальную точку и точку минимума;

3) теоретическое описание базового метода безусловной минимизации функций многих переменных;

4) алгоритм базового метода многомерной безусловной миними-

зации;

5) теоретическое описание метода одномерного поиска в многомерном пространстве;

6) алгоритм метода одномерного поиска;

7) распечатки используемых компьютерных программ;

8) табличное представление процесса минимизации и его итоги – количество вычислений значений целевой функции, минимальное значение функции и соответствующее ему значение точки минимума, достигнутую точность;

9) трехмерный график целевой функции и траектория поиска со всеми точками;

10) двумерный график линий уровня целевой функции и траектория поиска со всеми точками;

11) анализ полученных результатов;

12) выводы о проделанной лабораторной работе.

Контрольные вопросы

1. Дайте определения глобального и локального минимумов и максимумов функции многих переменных.

2. Приведите определение многоэкстремальной функции.

3. Дайте определение унимодальной функции.

4. Сформулируйте задачу многомерной безусловной минимизации.

5. Дайте определение решения задачи многомерной безусловной минимизации.

6. Какие функции называются выпуклыми и строго выпуклыми?

7. Какие функции называются вогнутыми и строго вогнутыми?

8. Объясните геометрический смысл определений выпуклой и вогнутой функций.

9. Сформулируйте и докажите теорему о минимуме выпуклой функции.

10. Сформулируйте следствия теоремы о минимуме выпуклой

функции.

11. Какие методы применяются для решения задачи многомерной безусловной минимизации?

12. Дайте определение линии уровня и поверхности уровня функции многих переменных.

13. Раскройте роль условий оптимальности при изучении задач оптимизации.

14. Дайте определение градиента функции многих переменных.

15. Сформулируйте и докажите необходимое условие оптимальности первого порядка для существования экстремума функции нескольких переменных.

16. Дайте определение стационарной точки функции нескольких переменных.

17. Сформулируйте необходимое и достаточное условие оптимальности первого порядка для существования минимума выпуклой функции нескольких переменных.

18. Дайте определение матрицы Гессе функции многих переменных.

19. Объясните свойства матрицы Гессе функции многих переменных.

20. Сформулируйте и докажите необходимое условие оптимальности второго порядка для существования минимума функции многих переменных.

21. Сформулируйте и докажите достаточное условие оптимальности второго порядка для существования минимума функции многих переменных.

22. Сформулируйте критерий Сильвестра – Якоби для исследования матрицы Гессе на положительную и отрицательную определенность.

23. Как исследовать функцию многих переменных на экстремум с помощью критерия Сильвестра – Якоби?

24. Опишите метод циклического покоординатного спуска.

25. По какому принципу задается направление шага в методе цик-

лического покоординатного спуска?

26. Как вычисляется величина шага в методе циклического покоординатного спуска?

27. Как заканчиваются вычисления в методе циклического покоординатного спуска?

28. Какого порядка метод циклического покоординатного спуска?

29. Составьте алгоритм метода циклического покоординатного спуска.

30. Укажите достоинства и недостатки метода циклического покоординатного спуска.

31. Дайте определение направления спуска.

32. Какие методы многомерной минимизации функций называются методами спуска?

33. Запишите условие точного одномерного поиска в многомерном пространстве.

34. Приведите формулы итерации метода спуска.

35. Составьте алгоритм метода спуска.

36. Как выполняется масштабирование направления поиска в методе спуска?

37. Дайте определение направления наискорейшего спуска.

38. Опишите метод наискорейшего спуска.

39. По какому принципу задается направление шага в методе наискорейшего спуска?

40. Как вычисляется величина шага в методе наискорейшего спуска?

41. Как заканчиваются вычисления в методе наискорейшего спуска?

42. Какого порядка метод наискорейшего спуска?

43. Составьте алгоритм метода наискорейшего спуска.

44. Какова особенность траектории поиска для метода наискорейшего спуска?

45. Укажите достоинства и недостатки метода наискорейшего спуска.

46. Зачем нужно уметь вычислять градиент функции многих переменных?

47. Укажите способы вычисления градиента функции многих переменных.

48. Запишите формулу конечной разности вперёд для вычисления проекций градиента функции многих переменных.

49. Какова погрешность формулы конечной разности вперёд для вычисления проекций градиента функции многих переменных?

50. Как следует задавать приращение аргумента в формуле конечной разности вперёд для вычисления проекций градиента функции многих переменных?

51. Составьте алгоритм вычисления градиента функции многих переменных.

52. Какое количество дополнительных вычислений функции многих переменных необходимо выполнить для вычисления градиента по формуле конечной разности вперёд?

53. Запишите формулу центральной разности для вычисления проекций градиента функции многих переменных.

54. Какова погрешность формулы центральной разности для вычисления проекций градиента функции многих переменных?

55. Какое количество дополнительных вычислений функции многих переменных необходимо выполнить для вычисления градиента по формуле центральной разности?

2. МЕТОД НЬЮТОНА И ЕГО МОДИФИКАЦИИ

Данный раздел посвящен методам второго порядка для безусловной минимизации функции многих переменных, которые основаны на формуле Ньютона с использованием матрицы вторых частных производных целевой функции – матрицы Гессе. Обосновывается классический метод Ньютона, анализируются его достоинства и недостатки. Рассматривается метод Ньютона с одномерным поиском. Представляется метод Ньютона с заданием направления спуска. Обосновывается метод Марквардта и его модификация с одномерным поиском. Приводятся алгоритмы для рассмотренных методов. Выводятся формулы для вычисления матрицы Гессе функции многих переменных. Дается описание лабораторной работы по разработке подпрограмм для модификаций метода Ньютона.

2.1. Метод Ньютона

Основная идея метода Ньютона заключается в итеративном использовании квадратичной аппроксимации целевой функции в текущей точке поиска и минимизации этой аппроксимации. Разложим дважды дифференцируемую целевую функцию $f(\mathbf{x})$ в ряд Тейлора в фиксированной точке \mathbf{x} при произвольном приращении аргумента $\Delta\mathbf{x}$, ограничиваясь слагаемыми второго порядка малости,

$$f(\mathbf{x} + \Delta\mathbf{x}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \Delta\mathbf{x} + \frac{1}{2} \Delta\mathbf{x}^T \nabla^2 f(\mathbf{x}) \Delta\mathbf{x} + o(\|\Delta\mathbf{x}\|^2).$$

Пренебрегая слагаемыми выше второго порядка малости и обозначая приращение аргумента $\mathbf{s} = \Delta\mathbf{x}$, градиент $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$, матрицу Гессе $\mathbf{H}(\mathbf{x}) = \nabla^2 f(\mathbf{x})$, получим квадратичную функцию

$$q(\mathbf{s}) = f(\mathbf{x}) + \mathbf{g}(\mathbf{x})^T \mathbf{s} + \frac{1}{2} \mathbf{s}^T \mathbf{H}(\mathbf{x}) \mathbf{s}.$$

Вычислим значение аргумента \mathbf{s} , которое минимизирует эту функцию. Используя формулы векторного дифференцирования

$$\nabla(\mathbf{a}^T \mathbf{x}) = \mathbf{a}, \quad \nabla(\mathbf{x}^T \mathbf{A} \mathbf{x}) = 2\mathbf{A} \mathbf{x},$$

запишем градиент квадратичной функции

$$\nabla q(\mathbf{s}) = \mathbf{g}(\mathbf{x}) + \mathbf{H}(\mathbf{x}) \mathbf{s}.$$

Учитывая необходимое условие минимума $\nabla q(\mathbf{s}) = \mathbf{0}$, получим систему линейных алгебраических уравнений (СЛАУ)

$$\mathbf{H}(\mathbf{x}) \mathbf{s} = -\mathbf{g}(\mathbf{x}). \quad (2.1)$$

Решая эту систему относительно вектора \mathbf{s} , найдем вектор перемещения в точку минимума квадратичной функции

$$\mathbf{s} = -\mathbf{H}(\mathbf{x})^{-1} \mathbf{g}(\mathbf{x}). \quad (2.2)$$

Метод минимизации функции, основанный на формулах (2.1) или (2.2), называется *методом Ньютона*. Формулы (2.1) и (2.2) можно представить в виде:

$$\nabla^2 f(\mathbf{x}) \Delta \mathbf{x} = -\nabla f(\mathbf{x}), \quad \Delta \mathbf{x} = -\nabla^2 f(\mathbf{x})^{-1} \nabla f(\mathbf{x}).$$

Метод Ньютона минимизирует положительно определенную квадратичную функцию за один шаг из любой начальной точки \mathbf{x}_0

$$\mathbf{x}^* = \mathbf{x}_0 - \mathbf{H}(\mathbf{x}_0)^{-1} \mathbf{g}(\mathbf{x}_0).$$

В случае же минимизации функции общего вида метод Ньютона применяется итерационно. Обозначая в текущей точке поиска \mathbf{x}_k значения градиента $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$ и матрицы Гессе $\mathbf{H}_k = \nabla^2 f(\mathbf{x}_k)$, получим на основании равенства (2.1) итерационные формулы метода Ньютона

для номеров итераций $k = 0, 1, 2, \dots$ из произвольной начальной точки поиска \mathbf{x}_0 :

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k, \quad \mathbf{H}_k \mathbf{s}_k = -\mathbf{g}_k. \quad (2.3)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность, $\varepsilon > 0$. По формулам (2.3) составим алгоритм метода Ньютона.

А л г о р и т м м е т о д а Н ь ю т о н а .

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $\mathbf{H} = \nabla^2 f(\mathbf{x})$.

2. Решить СЛАУ $\mathbf{H} \cdot \mathbf{s} = -\mathbf{g}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.

4. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 1.

5. Остановиться.

В этом алгоритме на шаге 2 решается система линейных алгебраических уравнений одним из стандартных методов используемой вычислительной системы. Например, в системе MATLAB СЛАУ решается оператором $\mathbf{s} = -\mathbf{H} \backslash \mathbf{g}$. Здесь применяется формула (2.1) метода Ньютона. На шаге 2 возможно применение и формулы (2.2), в которой используется обратная матрица Гессе

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}_k^{-1} \mathbf{g}_k. \quad (2.4)$$

Однако обращение матрицы в формуле (2.4) требует выполнения гораздо большего количества операций, чем решение СЛАУ в (2.3).

Пр и м е р 2.1. На рис. 2.1 представлена траектория поиска минимума квадратичной функции (1.3) методом Ньютона. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил

3 итерации и выполнил 19 вычислений функции. Поскольку определение матрицы Гессе в начальной точке поиска и решение СЛАУ (2.1) выполняется с вычислительными погрешностями, то на первой итерации найдено приближение точки минимума с погрешностью $5,3 \cdot 10^{-3}$. На последующих двух итерациях положение точки минимума уточнено и достигнута конечная погрешность $7,3 \cdot 10^{-23}$. Сравнивая эти результаты с результатами примеров 1.5 и 1.7, можно убедиться в преимуществе метода Ньютона перед методами циклического покоординатного спуска и наискорейшего спуска при минимизации квадратичной целевой функции.

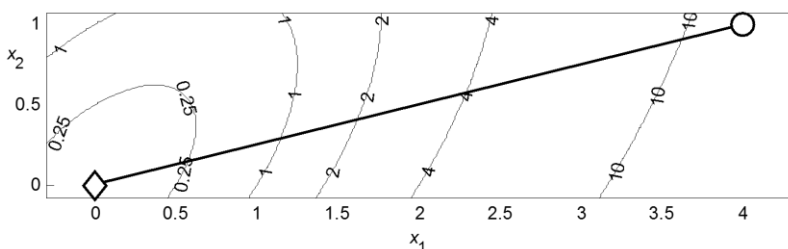


Рис. 2.1. Минимизация квадратичной функции методом Ньютона

Пример 2.2. На рис. 2.2 представлена траектория поиска минимума функции Розенброка методом Ньютона. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил 9 итераций и выполнил 55 вычислений функции. Сопоставление этих результатов с результатами примеров 1.6 и 1.8 также подтверждает преимущество метода Ньютона по сравнению с методами циклического покоординатного спуска и наискорейшего спуска.

В отличие от формул (1.20) метода наискорейшего спуска, в которых антиградиент $-\mathbf{g}_k$ задает только направление поиска, в формулах (2.3) метода Ньютона вектор \mathbf{s}_k задает и направление, и шаг перехода в следующую точку поиска.

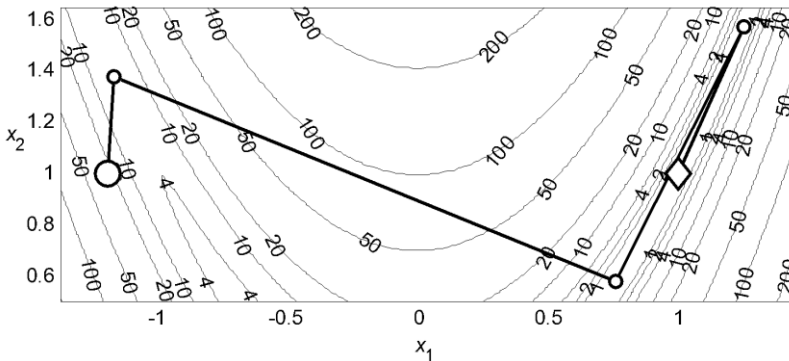


Рис. 2.2. Минимизация функции Розенброка методом Ньютона

Умножая равенство (2.2) скалярно на градиент $\mathbf{g}(\mathbf{x})$, получим:

$$\mathbf{g}(\mathbf{x})^T \mathbf{s} = -\mathbf{g}(\mathbf{x})^T \mathbf{H}(\mathbf{x})^{-1} \mathbf{g}(\mathbf{x}).$$

Если матрица Гессе $\mathbf{H}(\mathbf{x})$ положительно определенная, то и обратная к ней матрица $\mathbf{H}(\mathbf{x})^{-1}$ также положительно определенная. В этом случае $\mathbf{g}(\mathbf{x})^T \mathbf{s} < 0$ и вектор \mathbf{s} определяет направление спуска. Если же матрица $\mathbf{H}(\mathbf{x})$ отрицательно определенная, то вектор \mathbf{s} определяет направление возрастания функции. В этом случае вектор \mathbf{s} приводит в точку максимума квадратичной функции. Условию $\nabla q(\mathbf{s}) = \mathbf{0}$ отвечает также седловая точка. Поэтому одним из недостатков метода Ньютона является возможное возрастание значений функции $f(\mathbf{x}_{k+1}) > f(\mathbf{x}_k)$ и расходимость метода для функции общего вида. Другим недостатком является использование матрицы вторых частных производных, требующее дополнительных вычислений.

Метод Ньютона также называют *методом Ньютона – Рафсона*. Так как он использует вторые производные целевой функции, это метод второго порядка.

2.3. Метод Ньютона с одномерным поиском

При минимизации произвольной целевой функции $f(\mathbf{x})$ формулы метода Ньютона (2.3), основанные на аппроксимации функции квадратичной функцией в малой окрестности текущей точки поиска, могут приводить к возрастанию значений функции. Для повышения надежности метода Ньютона и обеспечения убывания значений целевой функции применяется одномерная минимизация функции в направлении поиска \mathbf{s} , найденном по формулам (2.1) или (2.2). На каждой итерации такого модифицированного метода Ньютона из точки \mathbf{x}_k выполняется одномерный поиск в направлении метода Ньютона $\mathbf{d}_k = \mathbf{s}_k$

$$\lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k).$$

Таким образом, получим формулы метода Ньютона с одномерным поиском:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \mathbf{H}_k \mathbf{d}_k = -\mathbf{g}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k). \quad (2.5)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность, $\varepsilon > 0$. По формулам (2.5) составим алгоритм метода Ньютона с одномерным поиском.

Алгоритм метода Ньютона с одномерным поиском.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $\mathbf{H} = \nabla^2 f(\mathbf{x})$.
2. Решить СЛАУ $\mathbf{H} \cdot \mathbf{d} = -\mathbf{g}$.

3. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

4. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.

5. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 1.

6. Остановиться.

По сравнению с предыдущим алгоритмом в этом алгоритме добавлен шаг 3, на котором выполняется одномерный поиск. Шаг перехода в следующую точку поиска обозначен через \mathbf{s} . Итерации продолжаются, пока норма \mathbf{s} больше допустимой погрешности.

Пример 2.3. Для нахождения точки минимума квадратичной функции (1.3) с допустимой погрешностью 10^{-3} метод Ньютона с одномерным поиском использовал 3 итерации и выполнил 28 вычислений функции. Траектория поиска минимума этим методом такая же, как и на рис. 2.1. На первой итерации найдено приближение точки минимума с погрешностью $4,8 \cdot 10^{-3}$. На последующих двух итерациях лучшие точки итераций приблизились к точке минимума и достигнутая конечная погрешность составила $7,2 \cdot 10^{-17}$.

Пример 2.4. На рис. 2.3 представлена траектория поиска минимума функции Розенброка методом Ньютона с одномерным поиском. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил 12 итераций и выполнил 260 вычислений функции.

Метод Ньютона с одномерным поиском надежнее исходного метода Ньютона. Однако его эффективность существенно зависит от того, является ли направление поиска направлением спуска.

2.3. Метод Ньютона с направлением спуска

Для повышения эффективности метода Ньютона необходимо обеспечить такое направление одномерного поиска, которое гарантированно приводит к убыванию функции, то есть является направлением спуска. Два распространенных случая, в которых метод Ньютона приводит к ошибочным направлениям, связаны с точками максимума целевой функции и седловыми точками. В этих и других случаях

необходимо проверять, является ли направление \mathbf{s}_k метода Ньютона направлением спуска. Если \mathbf{s}_k — направление спуска, то для него $\mathbf{s}_k^T \mathbf{g}_k < 0$, то есть скалярное произведение вектора направления \mathbf{s}_k из точки \mathbf{x}_k и вектора градиента в этой точке $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$ отрицательно. В геометрическом смысле это означает, что векторы \mathbf{s}_k и \mathbf{g}_k образуют тупой угол θ . Действительно, если $\mathbf{s}_k^T \mathbf{g}_k = |\mathbf{s}_k| \cdot |\mathbf{g}_k| \cdot \cos \theta$, то из неравенства $\mathbf{s}_k^T \mathbf{g}_k < 0$ следует $\cos \theta < 0$, то есть $\pi/2 < \theta \leq \pi$.

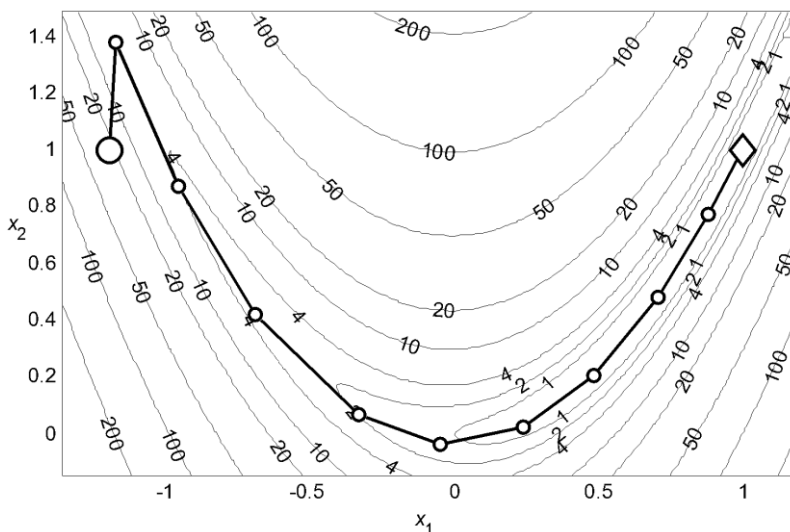


Рис. 2.3. Минимизация функции Розенброка методом Ньютона с одномерным поиском

Если условие $\mathbf{s}_k^T \mathbf{g}_k < 0$ не выполняется, то вектор \mathbf{s}_k не является направлением спуска. В этом случае для задания направления спуска целесообразно использовать антиградиент $-\mathbf{g}_k$. Таким образом, направление спуска в модифицированном методе Ньютона можно задать формулами:

$$\mathbf{H}_k \mathbf{s}_k = -\mathbf{g}_k; \quad \mathbf{d}_k = \begin{cases} \mathbf{s}_k, & \mathbf{s}_k^T \mathbf{g}_k < 0; \\ -\mathbf{g}_k, & \mathbf{s}_k^T \mathbf{g}_k \geq 0. \end{cases} \quad (2.6)$$

Вычислительные эксперименты показывают, что для функции общего вида наибольшее уменьшение значений функции на первой итерации обеспечивается в направлении наискорейшего спуска. Поэтому первую итерацию модифицированного метода Ньютона из заданной начальной точки \mathbf{x}_0 целесообразно проводить в направлении антиградиента $-\mathbf{g}_k = -\nabla f(\mathbf{x}_k)$ с использованием одномерного поиска:

$$\mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0, \quad \mathbf{d}_0 = -\mathbf{g}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0). \quad (2.7)$$

Последующие итерации проводятся по формулам:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad (2.8)$$

где направление поиска \mathbf{d}_k определяется формулой (2.6). Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε — допустимая погрешность, $\varepsilon > 0$. По формулам (2.6)–(2.8) составим алгоритм метода Ньютона с направлением спуска.

Алгоритм метода Ньютона с направлением спуска.

Входные параметры: \mathbf{x} — начальная точка поиска, $f(\mathbf{x})$ — процедура вычисления функции, ε — допустимая погрешность.

Выходной параметр \mathbf{x} — конечная точка поиска.

1. Вычислить $\mathbf{d} = -\nabla f(\mathbf{x})$.
2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.
3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.
4. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $\mathbf{H} = \nabla^2 f(\mathbf{x})$.

5. Решить СЛАУ $\mathbf{H} \cdot \mathbf{s} = -\mathbf{g}$.
6. Если $\mathbf{s}^T \cdot \mathbf{g} < 0$, то положить $\mathbf{d} = \mathbf{s}$, иначе положить $\mathbf{d} = -\mathbf{g}$.
7. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.
8. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.
9. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 4.
10. Остановиться.

По сравнению с предыдущим алгоритмом метода Ньютона в этом алгоритме добавлены шаги 1–3, на которых выполняется начальный одномерный поиск в направлении антиградиента, и шаг 6, на котором задается направление спуска.

Пример 2.5. На рис. 2.4 показана траектория минимизации квадратичной функции (1.3) модифицированным методом Ньютона с направлением спуска. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод использовал 3 итерации и выполнил 26 вычислений функции.

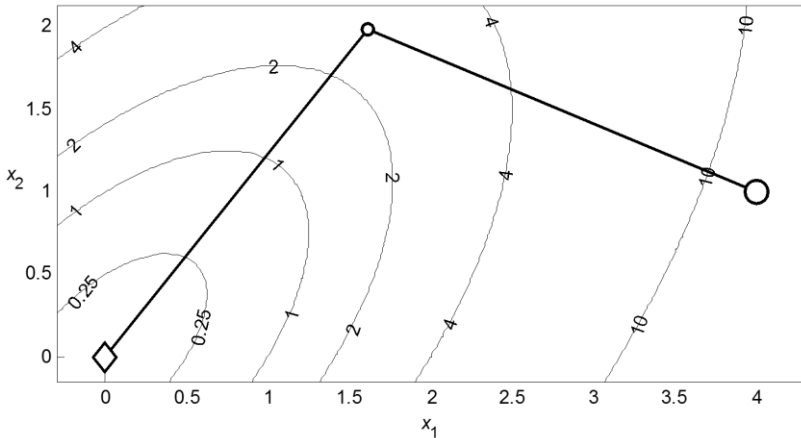


Рис. 2.4. Минимизация квадратичной функции методом Ньютона с направлением спуска

Пример 2.6. На рис. 2.5 представлена траектория поиска минимума функции Розенброка методом Ньютона с направлением спуска. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил 12 итераций и 307 вычислений функции.

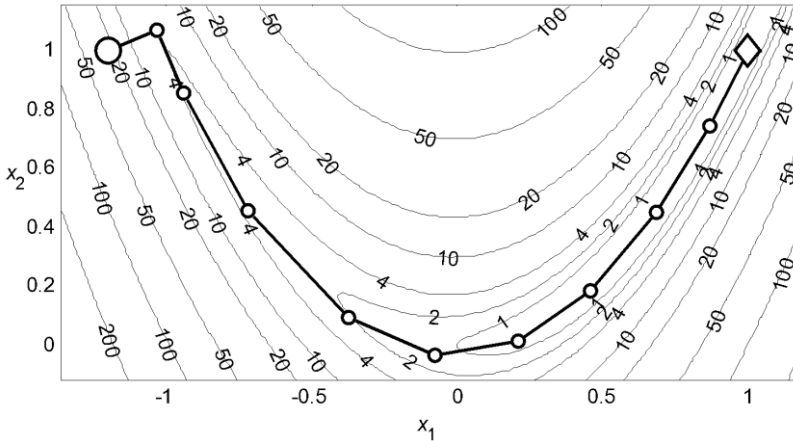


Рис. 2.5. Минимизация функции Розенброка методом Ньютона с направлением спуска

Метод Ньютона с одномерным поиском и заданием направления спуска надежнее предыдущих вариантов метода Ньютона.

2.4. Метод Марквардта

Этот метод является комбинацией методов наискорейшего спуска и Ньютона. Выше было отмечено, что движение в направлении антиградиента из начальной точки поиска \mathbf{x}_0 , расположенной на значительном расстоянии от точки минимума \mathbf{x}^* , обычно приводит к существенному уменьшению целевой функции. С другой стороны, направления эффективного поиска в окрестности точки минимума определяются по методу Ньютона.

Идея объединения методов наискорейшего спуска и Ньютона представлена американским математиком Д. Марквардтом в 1963 г. Он вместо формулы метода Ньютона (2.1) использовал формулу

$$[\mathbf{H}(\mathbf{x}) + \alpha \mathbf{E}]\mathbf{s} = -\mathbf{g}(\mathbf{x}), \quad (2.9)$$

где α – скалярный параметр, \mathbf{E} – единичная матрица той же размерности, что и $\mathbf{H}(\mathbf{x})$. При большом значении α матрицей $\mathbf{H}(\mathbf{x})$ в левой части уравнения (2.9) можно пренебречь, тогда получим уравнение $\alpha \mathbf{E}\mathbf{s} = -\mathbf{g}(\mathbf{x})$. В этом случае направление вектора $\mathbf{s} = -\mathbf{g}(\mathbf{x})/\alpha$ совпадает с антиградиентом, то есть направлением наискорейшего спуска. Если же $\alpha \rightarrow 0$, то в уравнении (2.9) можно пренебречь слагаемым $\alpha \mathbf{E}$. Тогда (2.9) превращается в (2.1) и \mathbf{s} становится шагом метода Ньютона. При промежуточном значении α направление вектора \mathbf{s} , удовлетворяющего (2.9), лежит между направлением антиградиента и направлением метода Ньютона. На этом свойстве уравнения (2.9) и основан метод Марквардта.

Начальное значение α_0 выбирается достаточно большим, $\alpha_0 \gg 1$, а затем в процессе поиска α_k уменьшается посредством умножения на коэффициент $\beta \in (0; 1)$. Таким образом, на начальных этапах выполняются шаги метода наискорейшего спуска $\mathbf{s}_k = -\mathbf{g}(\mathbf{x}_k)/\alpha_k$, а на конечных этапах используется шаг метода Ньютона. Формулы метода Марквардта можно представить в виде:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k, \quad (\mathbf{H}_k + \alpha_k \mathbf{E})\mathbf{s}_k = -\mathbf{g}_k, \quad \alpha_k = \alpha_{k-1}\beta. \quad (2.10)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность. Параметр метода Марквардта α_k позволяет не только изменять направление поиска, но и регулировать длину шага. Если шаг слишком большой и $f(\mathbf{x}_{k+1}) \geq f(\mathbf{x}_k)$, то пола-

гают $\alpha_k = \alpha_k / \beta$ и опять применяют формулы (2.10) уже с меньшим шагом s_k .

Формула Марквардта (2.9) позволяет устранить ещё один недостаток метода Ньютона, который связан с возможной плохой обусловленностью СЛАУ (2.1). *Плохо обусловленной* называется система уравнений, в которой малые погрешности исходных данных вызывают большие погрешности решения. *Числом обусловленности* матрицы \mathbf{H} называется произведение норм этой и обратной к ней матриц

$$\text{cond } \mathbf{H} = \|\mathbf{H}\| \cdot \|\mathbf{H}^{-1}\|.$$

Для единичной матрицы \mathbf{E} число обусловленности $\text{cond } \mathbf{E} = 1$, а для произвольной матрицы \mathbf{H} $\text{cond } \mathbf{H} \geq 1$. Чем больше число обусловленности матрицы, тем хуже обусловлена соответствующая СЛАУ. В частности, матрица \mathbf{H} может быть вырожденной, и тогда СЛАУ (2.1) не имеет решения в отличие от системы (2.9).

В точках поиска, далёких от минимума, матрица Гессе \mathbf{H} является, как правило, плохо обусловленной матрицей, но в этих точках для метода Марквардта $\alpha \gg 1$, поэтому $\mathbf{H} + \alpha \mathbf{E} \approx \alpha \mathbf{E}$, а, значит,

$$\text{cond } (\mathbf{H} + \alpha \mathbf{E}) \approx \text{cond } (\alpha \mathbf{E}) = \|\alpha \mathbf{E}\| \cdot \|(\alpha \mathbf{E})^{-1}\| = \alpha \cdot 1 \cdot \frac{1}{\alpha} \cdot 1 = 1.$$

Поэтому в методе Марквардта СЛАУ (2.9), определяющая направления поиска, хорошо обусловлена. Кроме того, поскольку единичная матрица \mathbf{E} положительно определена, то это свойство выполняется и для матрицы СЛАУ (2.9), так что найденное путем решения этой системы направление является направлением спуска.

По формулам (2.10) составим алгоритм метода Марквардта.

А л г о р и т м м е т о д а М а р к в а р д т а .

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, α_0 и β – параметры метода, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $f_{\mathbf{x}} = f(\mathbf{x})$.
2. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $\mathbf{H} = \nabla^2 f(\mathbf{x})$ и положить $\alpha = \alpha_0$.
3. Решить СЛАУ $(\mathbf{H} + \alpha \cdot \mathbf{E}) \cdot \mathbf{s} = -\mathbf{g}$.
4. Вычислить $\mathbf{y} = \mathbf{x} + \mathbf{s}$, $f_{\mathbf{y}} = f(\mathbf{y})$.
5. Если $f_{\mathbf{y}} \geq f_{\mathbf{x}}$, то положить $\alpha = \alpha/\beta$ и перейти к шагу 3.
6. Положить $\mathbf{x} = \mathbf{y}$, $f_{\mathbf{x}} = f_{\mathbf{y}}$, $\alpha_0 = \alpha_0 \cdot \beta$.
7. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 2.
8. Остановиться.

В этом алгоритме можно положить $\alpha_0 = 100$, $\beta = 0,5$. Алгоритм метода Марквардта характеризуется относительной простотой, свойством убывания значений целевой функции при переходе от итерации к итерации, высокой скоростью сходимости в окрестности точки минимума \mathbf{x}^* , а также отсутствием процедуры одномерного поиска.

Пример 2.7. На рис. 2.6 показана траектория минимизации квадратичной функции (1.3) методом Марквардта со всеми точками поиска. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод использовал 13 итераций и 79 вычислений функции.

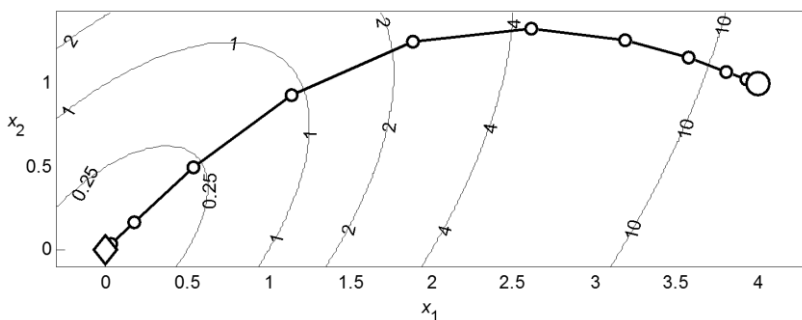


Рис. 2.6. Минимизация квадратичной функции методом Марквардта

Пример 2.8. На рис. 2.7 представлена траектория поиска минимума функции Розенброка методом Марквардта со всеми точками поиска. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод использовал 21 итерацию и 151 вычисление функции.

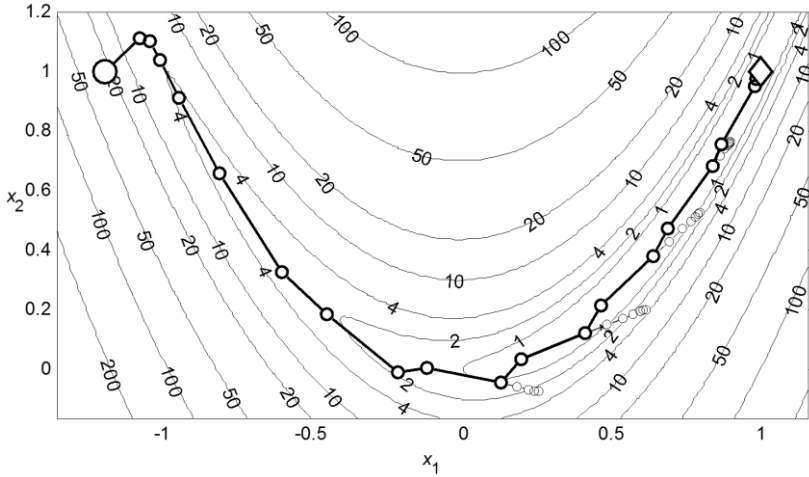


Рис. 2.7. Минимизация функции Розенброка методом Марквардта

В некоторых случаях применение одномерного поиска позволяет повысить надежность метода Марквардта. В методе Марквардта с одномерным поиском применяются формулы:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad (2.11)$$

$$(\mathbf{H}_k + \alpha_k \mathbf{E}) \mathbf{d}_k = -\mathbf{g}_k, \quad \alpha_k = \alpha_{k-1} \beta. \quad (2.12)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность. По формулам (2.11) и (2.12) составим алгоритм метода Марквардта с одномерным поиском.

Алгоритм метода Марквардта с одномерным поиском.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, α и β – параметры метода, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $\mathbf{H} = \nabla^2 f(\mathbf{x})$.
2. Решить СЛАУ $(\mathbf{H} + \alpha \cdot \mathbf{E}) \cdot \mathbf{d} = -\mathbf{g}$.
3. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.
4. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\alpha = \alpha \cdot \beta$.
5. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 1.
6. Остановиться.

В этом алгоритме можно положить $\alpha = 10^3$, $\beta = 0,5$.

Пример 2.9. На рис. 2.8 представлена траектория поиска минимума квадратичной функции (1.3) методом Марквардта с одномерным поиском. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил 11 итераций и 101 вычисление функции. Сравнивая результаты этого примера с результатами примера 1.7 для метода наискорейшего спуска, отметим уменьшение числа итераций за счет более эффективного направления поиска на завершающих итерациях из-за увеличения количества вычислений функции, связанного с нахождением матрицы Гессе. Траектории поиска на рис. 1.7 и рис. 2.8 выглядят одинаково, так как на начальных итерациях метода Марквардта преобладает направление наискорейшего спуска. Для квадратичных функций метод Маркварда использовать нецелесообразно.

Пример 2.10. На рис. 2.9 представлена траектория поиска минимума функции Розенброка методом Марквардта с одномерным поиском. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод использовал 15 итераций и 398 вычислений функции.

Здесь количество вычислений функции также оказалось больше, чем для всех предыдущих вариантов метода Ньютона.

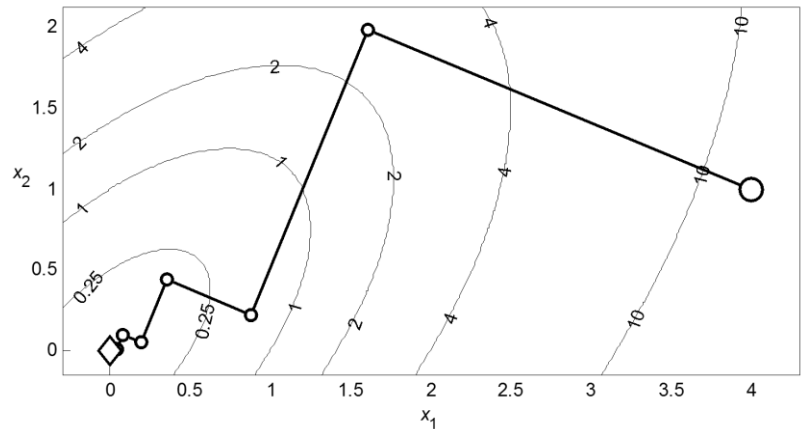


Рис. 2.8. Минимизация квадратичной функции методом Марквардта с одномерным поиском

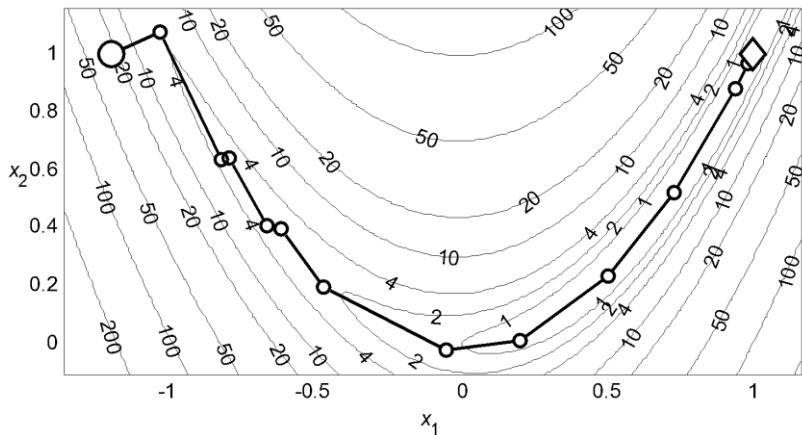


Рис. 2.9. Минимизация функции Розенброка методом Марквардта с одномерным поиском

Метод Марквардта широко используется при решении задач, в которых целевая функция записывается в виде суммы квадратов. Именно такая задача и рассматривалась Марквардтом. Метод Марквардта также называется *методом Левенберга – Марквардта*.

2.5. Вычисление матрицы Гессе

Для минимизации дважды дифференцируемой целевой функции $f(\mathbf{x})$, зависящей от n переменных x_1, x_2, \dots, x_n , все варианты метода Ньютона, которые являются методами второго порядка, используют матрицу Гессе вторых частных производных

$$\nabla^2(\mathbf{x}) = \mathbf{H}(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{pmatrix}.$$

По теореме Шварца

$$\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\mathbf{x})}{\partial x_j \partial x_i}, \quad i, j = \overline{1, n}, \quad i \neq j.$$

Поэтому матрица Гессе является симметрической матрицей, для которой симметричные относительно главной диагонали элементы совпадают: $H_{ij}(\mathbf{x}) = H_{ji}(\mathbf{x})$. В силу этого свойства при транспонировании матрица Гессе не изменяется $\mathbf{H}(\mathbf{x})^T = \mathbf{H}(\mathbf{x})$.

Во многих задачах оптимизации, которые встречаются на практике, целевые функции представляются сложными выражениями или вычисляются алгоритмически, поэтому матрица Гессе в методах второго порядка определяется численно с помощью конечных разностей.

Для вычисления матрицы Гессе целевую функцию $f(\mathbf{x})$ разложим в ряд Тейлора, учитывая слагаемые второго порядка малости:

$$f(\mathbf{x} + \Delta\mathbf{x}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \Delta\mathbf{x} + \frac{1}{2} \Delta\mathbf{x}^T \nabla^2 f(\mathbf{x}) \Delta\mathbf{x} + o(\|\Delta\mathbf{x}\|^2).$$

Вначале рассмотрим квадратичную функцию двух переменных при $\mathbf{x} = \mathbf{0}$:

$$f(\Delta\mathbf{x}) = c_0 + (c_1 \quad c_2) \begin{pmatrix} \Delta x_1 \\ \Delta x_2 \end{pmatrix} + \frac{1}{2} (\Delta x_1 \quad \Delta x_2) \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} \begin{pmatrix} \Delta x_1 \\ \Delta x_2 \end{pmatrix}. \quad (2.13)$$

Для такой функции

$$\mathbf{g}(\mathbf{0}) = \nabla f(\mathbf{0}) = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \quad \mathbf{H} = \nabla^2 f(\mathbf{0}) = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}.$$

В силу симметричности матрицы Гессе $c_{12} = c_{21}$. Поэтому функция (2.13) имеет шесть неизвестных параметров: c_0 , c_1 , c_2 , c_{11} , c_{12} , c_{22} . Эти параметры вычислим по шести значениям функции в различных ее точках (рис. 2.10). Положим $\Delta x_1 = \Delta x_2 = \delta$, зададим точки $(0, 0)$, $(\delta, 0)$, $(-\delta, 0)$, $(0, \delta)$, $(0, -\delta)$, (δ, δ) и вычислим

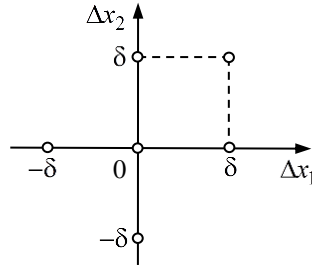


Рис. 2.10. Точки для вычислений

значения функции в этих точках: $f(0, 0)$, $f(\delta, 0)$, $f(-\delta, 0)$, $f(0, \delta)$, $f(0, -\delta)$, $f(\delta, \delta)$. Подставляя эти значения в уравнение (2.13), получим систему шести уравнений с шестью неизвестными, решая которую имеем:

$$c_0 = f(0, 0), \quad c_1 = \frac{f(\delta, 0) - f(-\delta, 0)}{2\delta}, \quad c_2 = \frac{f(0, \delta) - f(0, -\delta)}{2\delta},$$

$$c_{11} = \frac{f(\delta, 0) - 2f(0, 0) + f(-\delta, 0)}{\delta^2}, \quad c_{22} = \frac{f(0, \delta) - 2f(0, 0) + f(0, -\delta)}{\delta^2},$$

$$c_{12} = \frac{f(\delta, \delta) - f(\delta, 0) - f(0, \delta) + f(0, 0)}{\delta^2}.$$

Полученные для квадратичной функции (2.13) формулы обобщаются для вычисления градиента и матрицы Гессе произвольной функции n переменных с использованием ортов системы координат \mathbf{e}_i , $i = \overline{1, n}$.

Проекции градиента вычислим по формуле центральных разностей (1.22):

$$g_i(\mathbf{x}) = \frac{f(\mathbf{x} + \delta \mathbf{e}_i) - f(\mathbf{x} - \delta \mathbf{e}_i)}{2\delta}, \quad i = \overline{1, n}. \quad (2.14)$$

Диагональные элементы матрицы Гессе вычисляются по формуле конечной разности второго порядка:

$$H_{ii}(\mathbf{x}) = \frac{f(\mathbf{x} + \delta \mathbf{e}_i) - 2f(\mathbf{x}) + f(\mathbf{x} - \delta \mathbf{e}_i)}{\delta^2}, \quad i = \overline{1, n}. \quad (2.15)$$

Недиагональные элементы матрицы Гессе вычислим по формуле

$$H_{ij}(\mathbf{x}) = \frac{f(\mathbf{x} + \delta \mathbf{e}_i + \delta \mathbf{e}_j) - f(\mathbf{x} + \delta \mathbf{e}_i) - f(\mathbf{x} + \delta \mathbf{e}_j) + f(\mathbf{x})}{\delta^2}, \quad (2.16)$$

где $i = \overline{1, n-1}$, $j = \overline{i+1, n}$.

Формулы (2.14)–(2.16) кроме значения целевой функции в точке \mathbf{x} требуют $n(n+3)/2$ дополнительных вычислений значений функции. При этом точность вычисления градиента и матрицы Гессе существенно зависит от величины приращения аргумента δ , которую связывают со значением машинного эpsilon ε_m . Обычно полагают $\delta \geq 10^8 \varepsilon_m$. По формулам (2.14)–(2.16) составлен алгоритм вычисления градиента и матрицы Гессе для функции многих переменных.

Алгоритм вычисления градиента и матрицы Гессе.

Входные параметры: \mathbf{x} и $f_{\mathbf{x}}$ – точка, в которой вычисляется градиент, и значение в ней функции; $f(\mathbf{x})$ – процедура вычисления целевой функции; δ – параметр приращения аргумента.

Выходные параметры: \mathbf{g} – градиент, \mathbf{H} – матрица Гессе.

1. Положить $n = \dim \mathbf{x}$, $\mathbf{y} = \mathbf{x}$, $\mathbf{z} = \mathbf{x}$, $a = 1/(2\delta)$, $b = 1/\delta^2$, $i = 1$.

2. Вычислить $y_i = x_i + \delta$, $f_{\mathbf{y}} = f(\mathbf{y})$, $z_i = x_i - \delta$, $f_{\mathbf{z}} = f(\mathbf{z})$.

3. Вычислить $g_i = a \cdot (f_{\mathbf{y}} - f_{\mathbf{z}})$, $H_{ii} = b \cdot (f_{\mathbf{y}} - 2 \cdot f_{\mathbf{x}} + f_{\mathbf{z}})$.

4. Положить $y_i = x_i$, $z_i = x_i$, $F_i = f_{\mathbf{y}}$.

5. Если $i < n$, то положить $i = i + 1$ и перейти к шагу 2.

6. Положить $i = 1$.

7. Положить $y_i = x_i + \delta$, $j = i + 1$.

8. Вычислить $y_j = x_j + \delta$, $f_{\mathbf{y}} = f(\mathbf{y})$, $H_{ij} = b \cdot (f_{\mathbf{y}} - F_i - F_j + f_{\mathbf{x}})$.

9. Положить $H_{ji} = H_{ij}$, $y_j = x_j$.

10. Если $j < n$, то положить $j = j + 1$ и перейти к шагу 8.

11. Положить $y_i = x_i$.

12. Если $i < n - 1$, то положить $i = i + 1$ и перейти к шагу 7.

13. Остановиться.

В этом алгоритме на шагах 2–5 вычисляются градиент и диагональные элементы матрицы Гессе, а на шагах 6–12 вычисляются недиагональные элементы матрицы Гессе. В примерах 2.1–2.10 градиент и матрица Гессе вычислены по приведенному алгоритму со значением параметра $\delta = 10^{-6}$.

Лабораторная работа

Тема лабораторной работы.

Разработка подпрограмм для модификаций метода Ньютона.

Цель лабораторной работы.

Разработать подпрограмму модифицированного метода Ньютона для безусловной минимизации функций многих переменных.

Порядок выполнения лабораторной работы.

1. Получить у преподавателя вариант целевой функции $f(\mathbf{x})$ многих переменных с начальной точкой поиска минимума \mathbf{x}_0 и задание на программирование одного из модифицированных методов Ньютона:

- 1) метод Ньютона с одномерным поиском;
- 2) метод Ньютона с заданием направления спуска;
- 3) метод Марквардта с одномерным поиском.

2. Получить у преподавателя задание на программирование одного из методов одномерного поиска в многомерном пространстве:

- 1) метод дихотомии;
- 2) метод деления интервала пополам;
- 3) метод Фибоначчи;
- 4) метод золотого сечения;
- 5) метод адаптации шага;
- 6) метод квадратичной интерполяции с тремя точками;
- 7) метод квадратичной интерполяции с двумя точками;
- 8) метод секущих;
- 9) метод касательных;
- 10) метод кубической интерполяции с четырьмя точками;
- 11) метод кубической интерполяции с двумя точками.

3. Аналитически найти точку \mathbf{x}^* минимума заданной функции $f(\mathbf{x})$ и вычислить минимальное значение функции $f^* = f(\mathbf{x}^*)$.

4. Написать подпрограмму вычисления значений функции $f(\mathbf{x})$ с входным параметром \mathbf{x} и выходным параметром f , а также подпрограмму вычисления матрицы Гессе и градиента целевой функции.

5. Протестировать подпрограмму вычисления функции $f(\mathbf{x})$ при вычислении значений $f_0 = f(\mathbf{x}_0)$ и $f^* = f(\mathbf{x}^*)$, сравнить f^* с f_0 . Протестировать подпрограмму вычисления матрицы Гессе и градиента

целевой функции в точках \mathbf{x}_0 и \mathbf{x}^* .

6. На основании алгоритмов для заданного модифицированного метода Ньютона и метода одномерного поиска написать компьютерные подпрограммы в соответствии с требованиями:

1) входные параметры подпрограмм: \mathbf{x}_0 – начальная точка поиска, $f(\mathbf{x})$ – целевая функция, ε – допустимая погрешность;

2) выходные параметры подпрограмм: \mathbf{X} – массив всех точек поиска \mathbf{x}_k ; \mathbf{F} – массив соответствующих значений функции $f_k = f(\mathbf{x}_k)$; \mathbf{x} и $f_{\mathbf{x}}$ – лучшая точка поиска и значение в ней функции;

3) на каждой итерации выводить на экран компьютера строку, содержащую номер точки поиска k , шаг перехода в новую точку $h_k = \|\mathbf{s}_k\| = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|$, значение функции f_k и значение вектора переменных \mathbf{x}_k ;

4) итерации продолжаются до тех пор, пока выполняется неравенство $h_k > \varepsilon$;

5) после окончания процесса оптимизации на экране должна отображаться таблица, имеющая соответствующий заголовок и представляющая процесс минимизации функции;

6) под таблицей необходимо отобразить количество вычислений целевой функции, конечный шаг поиска, минимальное значение функции и соответствующую ему точку поиска.

7. Написать общую программу для выполнения вычислительного процесса минимизации целевой функции многих переменных и отображения на экране этого процесса в соответствии с требованиями:

1) за основу общей программы для минимизации функции взять общую программу из предыдущей лабораторной работы и модифицировать эту программу;

2) задать в программе значения \mathbf{x}_0 , h , ε ;

3) путем вызова разработанной подпрограммы заданного модифицированного метода Ньютона с использованием подпрограмм вы-

числения функции $f(\mathbf{x})$ и заданного метода одномерного поиска, а также значений \mathbf{x}_0 , h , ε реализовать итерационный процесс минимизации функции с построением таблицы процесса на экране и получением массивов значений вектора переменных \mathbf{X} и функции \mathbf{F} , лучшей точки поиска \mathbf{x} и значения в ней функции $f_{\mathbf{x}}$;

4) с помощью подпрограммы графического отображения итерационного процесса по массивам \mathbf{X} и \mathbf{F} отобразить графики процесса минимизации целевой функции многих переменных на экране.

8. Выполнить анализ полученных результатов:

1) на основании табличной и графической информации о работе метода одномерного поиска в многомерном пространстве определить конечные точки метода Свенна и показать найденный этим методом интервал неопределенности;

2) на нескольких начальных итерациях по табличным данным и графикам сопоставить начало работы модифицированного метода Ньютона с его рабочими формулами;

3) оценить эффективность модифицированного метода Ньютона и сопоставить ее с эффективностью методов многомерной безусловной оптимизации из предыдущей лабораторной работы.

9. Ответить на вопросы преподавателя по теме лабораторной работы.

10. Оформить и сдать отчет о проведенной лабораторной работе.

Содержание отчета:

1) титульный лист установленного образца с указанием организации, названия учебной дисциплины, темы работы, номера варианта, исполнителя и принимающего, города, года;

2) постановка задачи минимизации целевой функции, включающая задание функции, начальную точку и точку минимума;

3) теоретическое описание модифицированного метода Ньютона для безусловной минимизации функций многих переменных;

4) алгоритм модифицированного метода Ньютона;

5) теоретическое описание метода одномерного поиска в много-

мерном пространстве;

6) алгоритм метода одномерного поиска;

7) распечатки используемых компьютерных программ;

8) табличное представление процесса минимизации и его итоги – количество вычислений значений целевой функции, минимальное значение функции и соответствующее ему значение точки минимума, достигнутую точность;

9) трехмерный график функции и траектории поиска со всеми точками;

10) двумерный график линий уровня функции и траектории поиска со всеми точками;

11) анализ полученных результатов;

12) выводы о проделанной лабораторной работе.

Контрольные вопросы

1. В чем заключается основная идея метода Ньютона?

2. Представьте функцию многих переменных рядом Тейлора, ограничиваясь слагаемым второго порядка малости.

3. Выведите формулу метода Ньютона.

4. Дайте определение метода Ньютона для решения задачи безусловной минимизации функции многих переменных.

5. Приведите различные формулы метода Ньютона.

6. Запишите итерационные формулы метода Ньютона.

7. Опишите метод Ньютона.

8. Какой принцип задания направления шага в методе Ньютона?

9. Как вычисляется величина шага в методе Ньютона?

10. Как заканчиваются вычисления в методе Ньютона?

11. Какого порядка метод Ньютона?

12. Составьте алгоритм метода Ньютона.

13. В чем заключается преимущество решения системы линейных алгебраических уравнений перед обращением матрицы Гессе при вычислении направления поиска в методе Ньютона?

14. При каком условии направление поиска в методе Ньютона будет направлением спуска?
15. Какое другое название метода Ньютона вы знаете?
16. Укажите достоинства и недостатки метода Ньютона.
17. Обоснуйте метод Ньютона с одномерным поиском.
18. Запишите формулы метода Ньютона с одномерным поиском.
19. Опишите метод Ньютона с одномерным поиском.
20. Как находят шаг в методе Ньютона с одномерным поиском?
21. Как заканчиваются вычисления в методе Ньютона с одномерным поиском?
22. Какого порядка метод Ньютона с одномерным поиском?
23. Составьте алгоритм метода Ньютона с одномерным поиском.
24. Укажите достоинства и недостатки метода Ньютона с одномерным поиском.
25. Как выполняется проверка того, что направление метода Ньютона является направлением спуска?
26. По какому принципу задается направление одномерного поиска в методе Ньютона с направлением спуска?
27. Опишите метод Ньютона с направлением спуска.
28. Как вычисляется величина шага в методе Ньютона с направлением спуска?
29. Как заканчиваются вычисления в методе Ньютона с направлением спуска?
30. Какого порядка метод Ньютона с направлением спуска?
31. Составьте алгоритм метода Ньютона с направлением спуска.
32. Укажите достоинства и недостатки метода Ньютона с направлением спуска.
33. Приведите формулу метода Марквардта.
34. Какой принцип задания направления в методе Марквардта?
35. Как задаются параметры в методе Марквардта?
36. Как решается проблема плохой обусловленности системы линейных алгебраических уравнений в методе Марквардта?
37. Запишите итерационные формулы метода Марквардта.

38. Опишите метод Марквардта.
39. Как вычисляется величина шага в методе Марквардта?
40. Как заканчиваются вычисления в методе Марквардта?
41. Какого порядка метод Марквардта?
42. Составьте алгоритм метода Марквардта.
43. Укажите достоинства и недостатки метода Марквардта.
44. Какое другое название метода Марквардта вы знаете?
45. Обоснуйте метод Марквардта с одномерным поиском.
46. Запишите итерационные формулы метода Марквардта с одномерным поиском.
47. Опишите метод Марквардта с одномерным поиском.
48. Как находят шаг в методе Марквардта с одномерным поиском?
49. Как заканчивают метод Марквардта с одномерным поиском?
50. Какого порядка метод Марквардта с одномерным поиском?
51. Дайте алгоритм метода Марквардта с одномерным поиском.
52. Укажите достоинства и недостатки метода Марквардта с одномерным поиском.
53. Зачем вычислять матрицу Гессе функции многих переменных?
54. Дайте способ вычисления матрицы Гессе.
55. Запишите формулы конечных разностей для вычисления матрицы Гессе функции многих переменных.
56. Как следует задавать приращение аргумента в формулах конечных разностей для вычисления матрицы Гессе?
57. Какое количество дополнительных вычислений функции многих переменных необходимо выполнить для вычисления матрицы Гессе по формулам конечных разностей?
58. Запишите формулу центральных разностей для вычисления проекций градиента функции многих переменных.
59. Какова погрешность формулы центральных разностей для вычисления проекций градиента функции многих переменных?
60. Составьте алгоритм вычисления матрицы Гессе и градиента функции многих переменных.

3. МЕТОДЫ СОПРЯЖЕННЫХ НАПРАВЛЕНИЙ

Данный раздел посвящен методам сопряженных направлений для минимизации функции многих переменных. Рассматриваются свойства квадратичной функции, на основании которых конструируются методы сопряженных направлений. Вводится понятие сопряженных векторов, изучаются их свойства. Формулируется и доказывается теорема методов сопряженных направлений. Обосновывается метод Пауэлла, основанный на формировании сопряженных направлений. Формулируется и доказывается теорема методов сопряженных градиентов, даются общие свойства этих методов. Представляется метод Флетчера – Ривса, приводятся его достоинства и недостатки, рекомендуется повышение эффективности метода с помощью рестартов. Обосновывается метод Полака – Рибьера, объясняется его преимущество перед методом Флетчера – Ривса. Дается описание лабораторной работы по практическому изучению методов сопряженных направлений.

3.1. Свойства квадратичной функции

Рассмотрим положительно определенную квадратичную функцию

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c, \quad (3.1)$$

где \mathbf{x} – вектор-столбец переменных размера n , \mathbf{A} – симметрическая и положительно определенная квадратная матрица размерности $n \times n$, \mathbf{b} – вектор-столбец размера n , c – скаляр,

$$\mathbf{x}^T \mathbf{A} \mathbf{x} > 0, \quad \forall \mathbf{x} \neq \mathbf{0}. \quad (3.2)$$

При этом все собственные значения матрицы \mathbf{A} положительны и $\mathbf{A}^T = \mathbf{A}$. Градиент функции (3.1) $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$ имеет вид:

$$\mathbf{g}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b} . \quad (3.3)$$

Дифференцируя это равенство, получим матрицу Гессе квадратичной функции $\nabla^2 f(\mathbf{x}) = \mathbf{A}$. Из (3.3) следует, что для произвольного приращения аргумента квадратичной функции $\Delta\mathbf{x}$

$$\mathbf{g}(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{A}(\mathbf{x} + \Delta\mathbf{x}) + \mathbf{b} .$$

Вычитая из этого равенства равенство (3.3), получим

$$\mathbf{g}(\mathbf{x} + \Delta\mathbf{x}) - \mathbf{g}(\mathbf{x}) = \mathbf{A}\Delta\mathbf{x} .$$

Обозначая $\Delta\mathbf{g} = \mathbf{g}(\mathbf{x} + \Delta\mathbf{x}) - \mathbf{g}(\mathbf{x})$, это свойство приращения градиента квадратичной функции запишем в виде

$$\Delta\mathbf{g} = \mathbf{A}\Delta\mathbf{x} . \quad (3.4)$$

Пусть \mathbf{x}^* – точка минимума квадратичной функции (3.1), и по необходимому условию минимума $\mathbf{g}^* = \mathbf{g}(\mathbf{x}^*) = \nabla f(\mathbf{x}^*) = \mathbf{0}$. Тогда по свойству (3.4) получим $\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{x}^*) = \mathbf{A}(\mathbf{x} - \mathbf{x}^*)$, то есть

$$\mathbf{g}(\mathbf{x}) = \mathbf{A}(\mathbf{x} - \mathbf{x}^*) . \quad (3.5)$$

Поскольку положительно определенная квадратичная функция является строго выпуклой, то необходимое условие минимума $\nabla f(\mathbf{x}) = \mathbf{0}$ для нее является одновременно и достаточным, поэтому для точки минимума \mathbf{x}^* из (3.3) получим уравнение $\mathbf{A}\mathbf{x}^* = -\mathbf{b}$, откуда

$$\mathbf{x}^* = -\mathbf{A}^{-1}\mathbf{b} .$$

При этом значении аргумента функция (3.1) принимает минимальное значение

$$f^* = f(\mathbf{x}^*) = \frac{1}{2}\mathbf{b}^T \mathbf{A}^{-1} \mathbf{A} \mathbf{A}^{-1} \mathbf{b} - \mathbf{b}^T \mathbf{A}^{-1} \mathbf{b} + c = \frac{1}{2}\mathbf{b}^T \mathbf{A}^{-1} \mathbf{b} - \mathbf{b}^T \mathbf{A}^{-1} \mathbf{b} + c ,$$

то есть

$$f^* = c - \frac{1}{2} \mathbf{b}^T \mathbf{A}^{-1} \mathbf{b}.$$

Пусть поиск минимума функции (3.1) проводится по итерационной схеме метода спуска

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad (3.6)$$

где λ_k вычисляется путем точной одномерной минимизации функции $\varphi(\lambda) = f(\mathbf{x}_k + \lambda \mathbf{d}_k)$ по параметру λ из точки \mathbf{x}_k в направлении вектора \mathbf{d}_k . Поскольку $\varphi'(\lambda) = \mathbf{g}(\mathbf{x}_k + \lambda \mathbf{d}_k)^T \mathbf{d}_k$, то при $\lambda = \lambda_k$ и $\varphi'(\lambda_k) = 0$ получим $\mathbf{g}(\mathbf{x}_k + \lambda_k \mathbf{d}_k)^T \mathbf{d}_k = 0$, то есть $\mathbf{g}(\mathbf{x}_{k+1})^T \mathbf{d}_k = 0$. Обозначим $\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k)$. Условие точного одномерного поиска принимает вид условия ортогональности

$$\mathbf{g}_{k+1}^T \mathbf{d}_k = 0. \quad (3.7)$$

По формуле (3.4)

$$\mathbf{g}_{k+1} - \mathbf{g}_k = \mathbf{A}(\mathbf{x}_{k+1} - \mathbf{x}_k), \quad (3.8)$$

С учетом формулы (3.6) $\mathbf{g}_{k+1} - \mathbf{g}_k = \mathbf{A} \lambda_k \mathbf{d}_k$, то есть

$$\mathbf{g}_{k+1} = \mathbf{g}_k + \lambda_k \mathbf{A} \mathbf{d}_k. \quad (3.9)$$

Теперь условие ортогональности (3.7) дает $(\mathbf{g}_k + \lambda_k \mathbf{A} \mathbf{d}_k)^T \mathbf{d}_k = 0$, откуда $\mathbf{g}_k^T \mathbf{d}_k + \lambda_k \mathbf{d}_k^T \mathbf{A} \mathbf{d}_k = 0$. Поскольку в силу положительной определенности матрицы \mathbf{A} и ее свойства (3.2) $\mathbf{d}_k^T \mathbf{A} \mathbf{d}_k > 0$, то

$$\lambda_k = - \frac{\mathbf{g}_k^T \mathbf{d}_k}{\mathbf{d}_k^T \mathbf{A} \mathbf{d}_k}. \quad (3.10)$$

Это и есть условие точного одномерного поиска для квадратичной функции. Формулы (3.2)–(3.10) представляют важные свойства квадратичной функции (3.1), которые используются при построении эффективных методов оптимизации.

3.2. Сопряжённые векторы и их свойства

В эффективных методах оптимизации используется понятие сопряженных направлений.

Пусть \mathbf{A} – симметрическая и положительно определенная матрица размерности $n \times n$.

Определение 3.1. *Ненулевые векторы $\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k$ пространства R^n при $k < n$ называются \mathbf{A} -сопряженными или просто сопряженными, если для них*

$$\mathbf{d}_i^T \mathbf{A} \mathbf{d}_j = 0, \quad \forall i \neq j. \quad (3.11)$$

Если матрица \mathbf{A} равна единичной матрице \mathbf{E} , то $\mathbf{E} \mathbf{d}_j = \mathbf{d}_j$ и условие сопряженности принимает вид $\mathbf{d}_i^T \mathbf{d}_j = 0$. Это означает, что скалярное произведение векторов равно нулю. В этом случае условие сопряженности векторов эквивалентно условию их ортогональности $\mathbf{d}_i \perp \mathbf{d}_j$. В частности, собственные векторы матрицы \mathbf{A} являются сопряженными.

Лемма 3.1. *Если векторы $\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k$ \mathbf{A} -сопряженные, то они линейно независимы.*

Доказательство. Составим линейную комбинацию векторов и приравняем ее нулю

$$\alpha_0 \mathbf{d}_0 + \alpha_1 \mathbf{d}_1 + \dots + \alpha_k \mathbf{d}_k = \mathbf{0}. \quad (3.12)$$

Транспонируем это равенство и умножим его справа на $\mathbf{A} \mathbf{d}_i$, где \mathbf{d}_i – произвольный вектор из данной системы векторов. В силу определения сопряженности (3.11) k слагаемых левой части обращаются в нуль, и в результате имеем

$$\alpha_i \mathbf{d}_i^T \mathbf{A} \mathbf{d}_i = 0.$$

По свойству (3.2) положительно определенной квадратичной функции

$\mathbf{d}_i^T \mathbf{A} \mathbf{d}_i > 0$, поэтому $\alpha_i = 0$. Итак, равенство (3.12) может выполняться только при нулевых значениях коэффициентов $\alpha_0, \alpha_1, \dots, \alpha_k$, откуда и следует линейная независимость сопряженных векторов. \square

Количество сопряженных векторов не может быть больше, чем n .

С л е д с т в и е 3.1. *Сопряженные векторы $\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{n-1}$ образуют базис в пространстве R^n .*

Любой вектор пространства R^n можно разложить по базису из сопряженных векторов. Для положительно определенной квадратичной функции эффективный поиск минимума можно проводить в сопряженных направлениях. Это утверждение основано на свойствах сопряженных направлений.

Теорема 3.1. *Пусть \mathbf{d} – некоторое направление поиска в пространстве R^n , а \mathbf{u} и \mathbf{v} – две различные точки минимума квадратичной функции $f(\mathbf{x})$, полученные из двух точек \mathbf{u}_0 и \mathbf{v}_0 в направлении \mathbf{d} . Тогда направление $\mathbf{s} = \mathbf{u} - \mathbf{v}$ сопряжено к направлению \mathbf{d} .*

Доказательство. Если \mathbf{u} и \mathbf{v} – точки минимума квадратичной функции $f(\mathbf{x})$, полученные из двух точек \mathbf{u}_0 и \mathbf{v}_0 в направлении \mathbf{d} , то по свойству (3.7) точного одномерного поиска для квадратичной функции выполняются условия ортогональности:

$$\mathbf{d}^T \mathbf{g}(\mathbf{u}) = 0, \quad \mathbf{d}^T \mathbf{g}(\mathbf{v}) = 0.$$

Но по формуле (3.3) градиента квадратичной функции

$$\mathbf{g}(\mathbf{u}) = \mathbf{A} \mathbf{u} + \mathbf{b}, \quad \mathbf{g}(\mathbf{v}) = \mathbf{A} \mathbf{v} + \mathbf{b}.$$

Условия ортогональности градиентов направлению \mathbf{d} примут вид:

$$\mathbf{d}^T (\mathbf{A} \mathbf{u} + \mathbf{b}) = 0, \quad \mathbf{d}^T (\mathbf{A} \mathbf{v} + \mathbf{b}) = 0.$$

Вычитая эти равенства и используя свойства матриц, получим:

$$\mathbf{d}^T \mathbf{A} (\mathbf{u} - \mathbf{v}) = 0.$$

Обозначая $\mathbf{s} = \mathbf{u} - \mathbf{v}$, приходим к утверждению теоремы. \square

Доказанную теорему называют *свойством параллельного подпространства*. По свойству параллельного подпространства выполнение двух одномерных поисков из разных точек в направлении \mathbf{d} позволяет получить сопряженное к \mathbf{d} направление \mathbf{s} (рис. 3.1). Точка минимума \mathbf{x}^* квадратичной функции $f(\mathbf{x})$ при $n = 2$ может быть найдена одномерным поиском из точек \mathbf{u} или \mathbf{v} в направлении \mathbf{s} , то есть в результате проведения трех одномерных поисков.

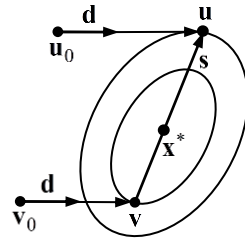


Рис. 3.1. Сопряженные направления

Гладкие овражные функции вблизи дна оврага можно аппроксимировать квадратичными функциями. Поэтому свойства сопряженных направлений позволяют находить эффективные направления поиска для любых гладких функций.

3.3. Теорема методов сопряжённых направлений

Методами сопряженных направлений для минимизации квадратичной функции n переменных называются итерационные методы с последовательным одномерным поиском в n сопряженных направлениях.

Теорема 3.2. *Метод сопряженных направлений из любой начальной точки находит минимум положительно определенной квадратичной функции не более чем за n итераций, причем в конечной точке любой итерации градиент ортогонален всем предыдущим направлениям поиска.*

Доказательство. Пусть для матрицы \mathbf{A} квадратичной функции (3.1) известны n сопряженных векторов $\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{n-1}$. Используем эти векторы как направления в задаче минимизации квад-

ратичной функции из произвольной точки $\mathbf{x}_0 \in R^n$ с помощью n точных одномерных поисков по формулам (3.6) и (3.10) при следующей схеме метода сопряженных направлений:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \lambda_k = -\frac{\mathbf{g}_k^T \mathbf{d}_k}{\mathbf{d}_k^T \mathbf{A} \mathbf{d}_k}, \quad k = \overline{0, n-1}. \quad (3.13)$$

Поскольку $\mathbf{x}_k = \mathbf{x}_{k-1} + \lambda_{k-1} \mathbf{d}_{k-1} = \mathbf{x}_{k-2} + \lambda_{k-2} \mathbf{d}_{k-2} + \lambda_{k-1} \mathbf{d}_{k-1}$, то, продолжая эти преобразования, получим:

$$\mathbf{x}_k = \mathbf{x}_{i+1} + \sum_{j=i+1}^{k-1} \lambda_j \mathbf{d}_j, \quad i = \overline{0, k-2}. \quad (3.14)$$

Отсюда и по формуле (3.3) имеем:

$$\begin{aligned} \mathbf{g}_k &= \mathbf{A} \mathbf{x}_k + \mathbf{b} = \mathbf{A} \left(\mathbf{x}_{i+1} + \sum_{j=i+1}^{k-1} \lambda_j \mathbf{d}_j \right) + \mathbf{b} = \mathbf{A} \mathbf{x}_{i+1} + \mathbf{b} + \mathbf{A} \sum_{j=i+1}^{k-1} \lambda_j \mathbf{d}_j, \\ \mathbf{g}_k &= \mathbf{g}_{i+1} + \sum_{j=i+1}^{k-1} \lambda_j \mathbf{A} \mathbf{d}_j. \end{aligned}$$

Тогда

$$\mathbf{d}_i^T \mathbf{g}_k = \mathbf{d}_i^T \left(\mathbf{g}_{i+1} + \sum_{j=i+1}^{k-1} \lambda_j \mathbf{A} \mathbf{d}_j \right) = \mathbf{d}_i^T \mathbf{g}_{i+1} + \sum_{j=i+1}^{k-1} \lambda_j \mathbf{d}_i^T \mathbf{A} \mathbf{d}_j, \quad i = \overline{0, k-2}.$$

В силу условия точного одномерного поиска (3.7) и определения сопряженных векторов (3.11) имеем $\mathbf{d}_i^T \mathbf{g}_k = 0$ при $i = \overline{0, k-2}$. Поскольку (3.7) можно представить в виде $\mathbf{d}_{k-1}^T \mathbf{g}_k = 0$, то для метода сопряженных направлений получим:

$$\mathbf{d}_i^T \mathbf{g}_k = 0, \quad i = \overline{0, k-1}. \quad (3.15)$$

Градиент \mathbf{g}_k ортогонален всем векторам направлений $\mathbf{d}_0, \mathbf{d}_1, \dots$,

\mathbf{d}_{k-1} . Тогда градиент \mathbf{g}_n ортогонален всем векторам сопряженных направлений $\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{n-1}$. Поскольку эти векторы образуют базис в пространстве R^n , то

$$\mathbf{g}_n = \alpha_0 \mathbf{d}_0 + \alpha_1 \mathbf{d}_1 + \dots + \alpha_{n-1} \mathbf{d}_{n-1}.$$

Умножая это векторное равенство скалярно на вектор \mathbf{g}_n и учитывая условия ортогональности (3.15), получим равенство $\mathbf{g}_n^T \mathbf{g}_n = 0$, откуда следует $\mathbf{g}_n = \mathbf{0}$. По свойству квадратичной функции (3.5) для градиента $\mathbf{g}_n = \mathbf{A}(\mathbf{x}_n - \mathbf{x}^*)$, то есть имеем $\mathbf{x}_n = \mathbf{x}^*$. Тем самым доказано, что метод сопряженных направлений (3.13) сходится к оптимальному решению задачи минимизации квадратичной функции не более чем за n итераций. \square

Равенства (3.15) показывают, что вектор градиента в конечной точке любой итерации ортогонален всем предыдущим направлениям поиска. Эти равенства отображают *принцип расширяющихся подпространств*, который допускает наглядную геометрическую интерпретацию, представленную на рис. 3.2.

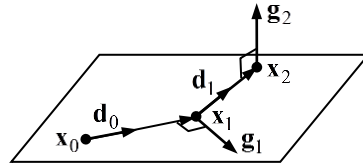


Рис. 3.2. Расширение подпространств

Таким образом, теорема 3.2 утверждает, что если для положительно определенной квадратичной функции n переменных заданы n взаимно сопряженных направлений $\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{n-1}$, то точка минимума этой функции \mathbf{x}^* может быть найдена из любой начальной точки \mathbf{x}_0 путем последовательного точного одномерного поиска вдоль этих n направлений.

3.4. Метод Пауэлла

Один из первых методов сопряженных направлений представлен английским математиком М. Д. Д. Пауэллом в 1964 году для минимизации функции нескольких переменных $f(\mathbf{x})$, $\mathbf{x} \in R^n$. Этот метод является развитием метода циклического покоординатного спуска, в котором последовательный одномерный поиск производится в направлениях ортов координатных осей $\mathbf{e}_1 = (1, 0, 0, \dots, 0)^T$, $\mathbf{e}_2 = (0, 1, 0, \dots, 0)^T$, ..., $\mathbf{e}_n = (0, 0, 0, \dots, 1)^T$. В отличие от метода покоординатного спуска в методе Пауэлла строится система n сопряженных направлений.

Каждая итерация метода Пауэлла начинается с определения n линейно независимых направлений $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n$. Первоначально эти направления совпадают с оортами координатных осей: $\mathbf{d}_i = \mathbf{e}_i$, $i = \overline{1, n}$.

Из начальной точки $\mathbf{u}_0 \in R^n$ производится вспомогательный одномерный поиск в последнем направлении \mathbf{d}_n :

$$\mathbf{x}_0 = \mathbf{u}_0 + \lambda_0 \mathbf{d}_n, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{u}_0 + \lambda \mathbf{d}_n). \quad (3.16)$$

Затем из точки \mathbf{x}_0 выполняется итерация метода покоординатного поиска в виде проведения n последовательных одномерных поисков в направлениях ортов координатных осей:

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \lambda_k \mathbf{d}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_{k-1} + \lambda \mathbf{d}_k), \quad k = \overline{1, n}. \quad (3.17)$$

Последняя точка \mathbf{x}_n получена при минимизации функции из точки \mathbf{x}_{n-1} в направлении \mathbf{d}_n . Таким образом, точки \mathbf{x}_0 и \mathbf{x}_n найдены путем минимизации функции в одном и том же направлении \mathbf{d}_n . В случае квадратичной целевой функции по свойству параллельного подпространства направление $\mathbf{s} = \mathbf{x}_n - \mathbf{x}_0$ сопряжено к направлению \mathbf{d}_n . Из системы направлений $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n$ исключается первое направление

\mathbf{d}_1 и добавляется новое направление \mathbf{s} . При этом полагают $\mathbf{d}_i = \mathbf{d}_{i+1}$ для $i = \overline{1, n-1}$ и $\mathbf{d}_n = \mathbf{s}$. Для квадратичной функции после окончания первой итерации два последние направления поиска \mathbf{d}_{n-1} и \mathbf{d}_n сопряжены.

В качестве начальной точки для следующей итерации задают $\mathbf{u}_0 = \mathbf{x}_n$ и проводят следующую итерацию по тем же формулам (3.16) и (3.17). Итерации продолжаются до тех пор, пока норма вектора \mathbf{s} остается больше допустимой погрешности ε .

Таким образом, для квадратичной функции после k итераций метода Пауэлла $k+1$ последние направления поиска $\mathbf{d}_{n-k}, \dots, \mathbf{d}_n$, выбранных для $(k+1)$ -й итерации, будут взаимно сопряженными. После $n-1$ итераций все направления будут взаимно сопряженными, поэтому метод Пауэлла обеспечивает минимизацию квадратичной функции не более чем за n итераций. Так, для минимизации квадратичной функции с двумя переменными $n=2$ и достаточно выполнения двух итераций. При этом точка минимума будет найдена после выполнения четырех одномерных поисков.

Алгоритм метода Пауэлла.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Положить $n = \dim \mathbf{x}$, $\mathbf{D} = \mathbf{E}(n)$.
2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{D}_n)$.
3. Положить $\mathbf{x} = \mathbf{x} + r \cdot \mathbf{D}_n$, $\mathbf{y} = \mathbf{x}$, $i = 1$.
4. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{D}_i)$.
5. Положить $\mathbf{x} = \mathbf{x} + r \cdot \mathbf{D}_i$.
6. Если $i < n$, то положить $i = i + 1$ и перейти к шагу 4.
7. Положить $i = 1$.

8. Положить $\mathbf{D}_i = \mathbf{D}_{i+1}$.
9. Если $i < n - 1$, то положить $i = i + 1$ и перейти к шагу 8.
10. Вычислить $\mathbf{D}_n = \mathbf{x} - \mathbf{y}$.
11. Если $\|\mathbf{D}_n\| > \varepsilon$, то перейти к шагу 2.
12. Остановиться.

В этом алгоритме на шаге 1 формируется матрица направлений поиска \mathbf{D} как единичная матрица \mathbf{E} размера n . Шаги 2–11 составляют итерацию метода Пауэлла. На шаге 2 выполняется одномерный поиск из точки \mathbf{x} в направлении последнего столбца \mathbf{D}_n матрицы \mathbf{D} . На шагах 4–6 производится последовательный одномерный поиск в направлениях всех столбцов матрицы \mathbf{D} . На шагах 7–10 формируется новая матрица направлений исключением первого ее столбца и добавлением последнего столбца. На шаге 11 проверяется критерий окончания вычислений.

Пример 3.1. На рис. 3.3 представлена траектория поиска минимума квадратичной функции (1.3) методом Пауэлла, включающая все точки поиска. Лучшие точки итераций соединены жирной линией. Для вычисления точки минимума с допустимой погрешностью 10^{-3} метод затратил 3 итерации, причем значение функции $9,2 \cdot 10^{-33}$ было найдено за две итерации при погрешности вычисления точки минимума $1,2 \cdot 10^{-16}$, а последняя итерация использована для проверки критерия окончания вычислений. Значения целевой функции были вычислены 35 раз. Сравнивая эти результаты с результатами примера 1.5 для минимизации той же функции методом циклического покоординатного спуска, можно убедиться в преимуществе метода Пауэлла.

Пример 3.2. На рис. 3.4 представлена траектория минимизации функции Розенброка методом Пауэлла, включающая лучшие точки итераций. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил 12 итераций и 529 вычислений функции.

Метод Пауэлла позволяет минимизировать гладкие овражные функции. Его достоинством является отсутствие использования производных функции, поэтому он относится к методам нулевого порядка.

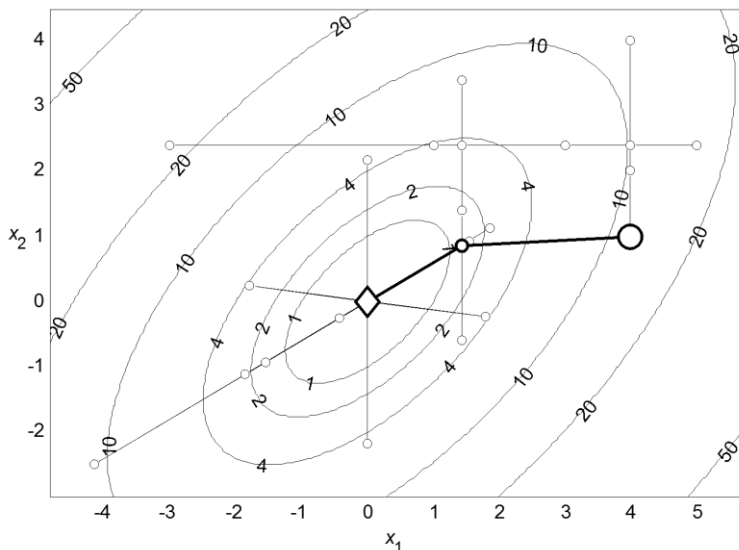


Рис. 3.3. Минимизация квадратичной функции методом Пауэлла

3.5. Методы сопряжённых градиентов

В методах сопряженных направлений сопряженные векторы как направления одномерного поиска можно определять разными способами.

Методами сопряженных градиентов для минимизации квадратичной функции $f(\mathbf{x})$ называются методы сопряженных направлений, в которых сопряженные векторы как направления одномерного поиска определяются на основании значений градиента целевой функции $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$.

Рассмотрим общие свойства методов сопряженных градиентов, в которых применяются формулы методов сопряженных направлений (3.13). Справедлива теорема методов сопряженных градиентов, где как и ранее $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$.

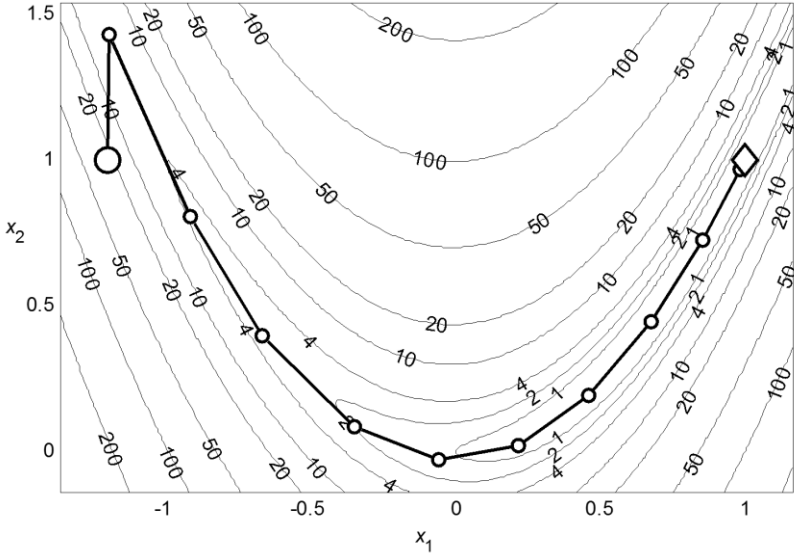


Рис. 3.4. Минимизация функции Розенброка методом Пауэлла

Теорема 3.3. Метод сопряженных градиентов является методом сопряженных направлений (3.13), в котором векторы направлений вычисляются по формулам

$$\mathbf{d}_0 = -\mathbf{g}_0, \quad \mathbf{d}_k = -\mathbf{g}_k + \beta_{k-1} \mathbf{d}_{k-1}, \quad \beta_{k-1} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}, \quad k = \overline{1, n-1}, \quad (3.18)$$

и выполняются свойства

$$\mathbf{g}_k^T \mathbf{g}_i = 0, \quad i = \overline{0, k-1}; \quad \mathbf{d}_k^T \mathbf{g}_k = -\mathbf{g}_k^T \mathbf{g}_k, \quad k = \overline{0, n-1}. \quad (3.19)$$

Доказательство. Пусть в методе сопряженных направлений (3.13) начальное направление поиска определяется по антиградиенту $\mathbf{d}_0 = -\mathbf{g}_0$ в начальной точке поиска $\mathbf{x}_0 \in R^n$, а последующие направления \mathbf{d}_k вычисляются как сумма антиградиента $-\mathbf{g}_k$ и линейной комбинации предыдущих направлений поиска:

$$\mathbf{d}_k = -\mathbf{g}_k + \sum_{i=0}^{k-1} \alpha_{ki} \mathbf{d}_i, \quad k = \overline{1, n-1}. \quad (3.20)$$

При этом коэффициенты α_{ki} выбирают таким образом, чтобы направления были сопряженными. При таком выборе (3.20) дает:

$$\mathbf{g}_k^T \mathbf{d}_i = \mathbf{g}_k^T \left(-\mathbf{g}_i + \sum_{j=0}^{i-1} \alpha_{ij} \mathbf{d}_j \right) = -\mathbf{g}_k^T \mathbf{g}_i + \sum_{j=0}^{i-1} \alpha_{ij} \mathbf{g}_k^T \mathbf{d}_j, \quad i = \overline{0, k-1}.$$

Отсюда с учетом (3.15) получим важное свойство методов сопряженных градиентов:

$$\mathbf{g}_k^T \mathbf{g}_i = 0, \quad i = \overline{0, k-1}. \quad (3.21)$$

Тем самым доказана первая группа равенств (3.19).

Умножим равенство (3.20) скалярно на вектор \mathbf{g}_k :

$$\mathbf{d}_k^T \mathbf{g}_k = -\mathbf{g}_k^T \mathbf{g}_k + \sum_{i=0}^{k-1} \alpha_{ki} \mathbf{d}_i^T \mathbf{g}_k, \quad k = \overline{1, n-1}.$$

По свойству методов сопряженных направлений (3.15) имеем:

$$\mathbf{d}_k^T \mathbf{g}_k = -\mathbf{g}_k^T \mathbf{g}_k, \quad k = \overline{0, n-1}. \quad (3.22)$$

Тем самым доказана вторая группа равенств (3.19).

Для определения коэффициентов α_{ki} в равенстве (3.20) умножим их скалярно на вектор $\mathbf{A} \mathbf{d}_j$ при $j = \overline{0, k-1}$:

$$\mathbf{d}_k^T \mathbf{A} \mathbf{d}_j = -\mathbf{g}_k^T \mathbf{A} \mathbf{d}_j + \sum_{i=0}^{k-1} \alpha_{ki} \mathbf{d}_i^T \mathbf{A} \mathbf{d}_j, \quad j = \overline{0, k-1}.$$

С использованием определения сопряженности (3.11) получим:

$$-\mathbf{g}_k^T \mathbf{A} \mathbf{d}_j + \alpha_{kj} \mathbf{d}_j^T \mathbf{A} \mathbf{d}_j = 0, \quad j = \overline{0, k-1}.$$

Умножая это равенство на λ_j и учитывая формулу (3.9) в виде $\lambda_j \mathbf{A} \mathbf{d}_j = \mathbf{g}_{j+1} - \mathbf{g}_j$, имеем:

$$-\mathbf{g}_k^T (\mathbf{g}_{j+1} - \mathbf{g}_j) + \alpha_{kj} \mathbf{d}_j^T (\mathbf{g}_{j+1} - \mathbf{g}_j) = 0, \quad j = \overline{0, k-1}.$$

Тогда с учетом свойств (3.21) и (3.22) получим:

$$\alpha_{kj} \mathbf{g}_j^T \mathbf{g}_j = 0, \quad j = \overline{0, k-2}; \quad -\mathbf{g}_k^T \mathbf{g}_k + \alpha_{kj} \mathbf{g}_j^T \mathbf{g}_j = 0, \quad j = k-1.$$

Отсюда

$$\alpha_{kj} = 0, \quad j = \overline{0, k-2}; \quad \alpha_{k, k-1} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}.$$

Обозначая $\beta_{k-1} = \alpha_{k, k-1}$, для формулы направления (3.20) окончательно имеем:

$$\mathbf{d}_k = -\mathbf{g}_k + \beta_{k-1} \mathbf{d}_{k-1}, \quad \beta_{k-1} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}, \quad k = \overline{1, n-1}. \quad (3.23)$$

Равенства (3.18) доказаны. Тем самым доказана теорема 3.3. \square

Для обеспечения сопряженности векторов направлений в методе сопряженных градиентов достаточно учитывать только предыдущий вектор направления. Формула для вычисления β_{k-1} в равенствах (3.23) представлена английскими математиками Р. Флетчером и К. М. Ривсом в 1964 году и называется *формулой Флетчера – Ривса*. Метод сопряженных градиентов обладает всеми свойствами методов сопряженных направлений и минимизирует квадратичную функцию при точном одномерном поиске не более чем за n итераций.

Важной особенностью формул (3.23) является то, что для построения вектора направления \mathbf{d}_k нужны лишь градиенты \mathbf{g}_k и \mathbf{g}_{k-1} в текущей и предыдущей точках соответственно, а также предыдущий вектор направления \mathbf{d}_{k-1} . Это обстоятельство оказывается особенно полезным для применения метода к гладким функциям общего вида.

3.6. Метод Флетчера – Ривса

Одним из самых известных методов сопряженных градиентов является *метод Флетчера – Ривса*. Приведем описание этого метода для минимизации произвольной целевой функции $f(\mathbf{x})$.

В методе Флетчера – Ривса направления одномерного поиска определяются на основании градиента целевой функции $\mathbf{g}(\mathbf{x}) = \nabla f(\mathbf{x})$. Задается начальная точка \mathbf{x}_0 , в ней вычисляется значение градиента $\mathbf{g}_0 = \nabla f(\mathbf{x}_0)$. В направлении антиградиента производится одномерный поиск и находится следующая точка:

$$\mathbf{d}_0 = -\mathbf{g}_0, \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0). \quad (3.24)$$

Последующие итерации основаны на вычислении градиента $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$, формулах методов спуска (1.16) и методов сопряженных градиентов (3.18):

$$\mathbf{d}_k = -\mathbf{g}_k + \beta_{k-1} \mathbf{d}_{k-1}, \quad \beta_{k-1} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}, \quad k = \overline{1, n-1}, \quad (3.25)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k). \quad (3.26)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность.

Вектор \mathbf{d}_k , построенный по формулам (3.25), для квадратичной функции является направлением спуска. Действительно, если $\mathbf{g}_k \neq 0$, то с учетом свойств (3.19) получим:

$$\mathbf{g}_k^T \mathbf{d}_k = -\mathbf{g}_k^T \mathbf{g}_k + \beta_{k-1} \mathbf{g}_k^T \mathbf{d}_{k-1} = -\|\mathbf{g}_k\|^2 < 0.$$

Заметим, что при $\beta_{k-1} = 0$ метод Флетчера – Ривса превращается в метод наискорейшего спуска.

По формулам (3.24)–(3.26) составим алгоритм метода Флетчера – Ривса.

Алгоритм метода Флетчера – Ривса.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, и положить $\mathbf{d} = -\mathbf{g}_x$.

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\mathbf{g}_y = \mathbf{g}_x$.

4. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, $\beta = \mathbf{g}_x^T \cdot \mathbf{g}_x / (\mathbf{g}_y^T \cdot \mathbf{g}_y)$.

5. Положить $\mathbf{d} = -\mathbf{g}_x + \beta \cdot \mathbf{d}$.

6. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 2.

7. Остановиться.

В этом алгоритме на шаге 2 выполняется одномерный поиск точки минимума из текущей точки поиска в направлении \mathbf{d} . Направление поиска вычисляется на шагах 4, 5. Для повышения эффективности одномерного поиска производят масштабирование направления поиска и используют вектор направления $\mathbf{d}^0 = \mathbf{d} / \|\mathbf{d}\|$ единичной длины. Шаг перехода в следующую точку поиска обозначен через \mathbf{s} . Итерации продолжаются, пока длина вектора \mathbf{s} больше заданной допустимой погрешности.

Достоинством метода Флетчера – Ривса является то, что его скорость сходимости значительно превышает скорость сходимости метода наискорейшего спуска. Он требует хранения в памяти малого количества информации в виде нескольких n -мерных векторов. Уменьшить объем памяти можно применением модификации предыдущего алгоритма.

Алгоритм 2 метода Флетчера – Ривса.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $q = \mathbf{g}^T \cdot \mathbf{g}$ и положить $\mathbf{d} = -\mathbf{g}$.

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$.

4. Вычислить $\mathbf{g} = \nabla f(\mathbf{x})$, $p = \mathbf{g}^T \cdot \mathbf{g}$, $\beta = p/q$.

5. Положить $\mathbf{d} = -\mathbf{g} + \beta \cdot \mathbf{d}$, $q = p$.

6. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 2.

7. Остановиться.

В отличие от предыдущего алгоритма здесь скалярные произведения градиентов хранятся в скалярных переменных p и q , что не требует дополнительного запоминания n -мерного вектора градиента. При решении реальных задач оптимизации значения градиента целевой функции вычисляется по формуле конечной разности вперед, что требует n добавочных вычислений функции. Относительно невысокий уровень требований к объему памяти компьютера делает метод Флетчера – Ривса особенно полезным при решении задач оптимизации большой размерности.

Пример 3.3. На рис. 3.5 представлена траектория минимизации квадратичной функции (1.3) методом Флетчера – Ривса, включающая лучшие точки итераций. Для вычисления точки минимума с допустимой погрешностью 10^{-3} метод затратил 3 итерации, причем значение функции $1,3 \cdot 10^{-14}$ найдено за две итерации при погрешности вычисления точки минимума $8,8 \cdot 10^{-8}$, а последняя итерация использована для проверки критерия окончания вычислений. Значения функции вычислены 19 раз. При этом траектория минимизации квадратичной функции методом Флетчера – Ривса такая же, как и траектория минимизации этой же функции модифицированным методом Ньютона

с направлением спуска из примера 2.5, представленная на рис. 2.4.

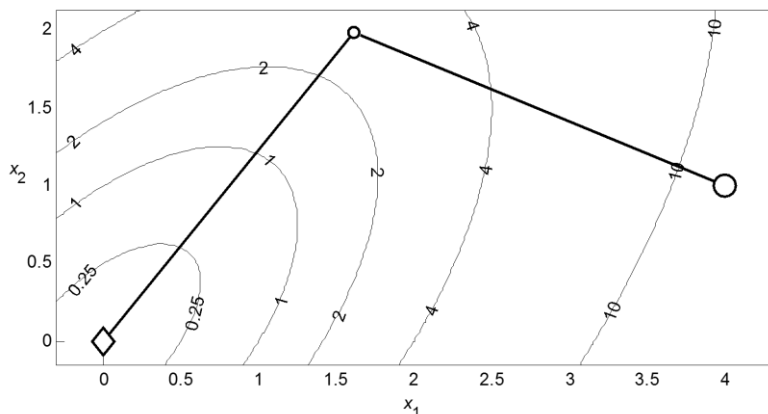


Рис. 3.5. Минимизация квадратичной функции методом Флетчера-Ривса

Пример 3.4. На рис. 3.6 представлена траектория поиска минимума функции Розенброка методом Флетчера – Ривса. До уменьшения шага с допустимой погрешностью 10^{-3} метод затратил 49 итераций и 461 вычисление функции. Метод проскочил мимо точки минимума, отмеченной звездочкой.

Недостатком метода Флетчера – Ривса является то, что он очень чувствителен к точности одномерного поиска, а также подвержен существенному влиянию ошибок машинных вычислений. При минимизации функций общего вида его эффективность может снижаться из-за нарушения условий сопряженности направлений поиска.

При нарушении условий сопряженности направлений метод Флетчера – Ривса начинает вычислять направления, движение вдоль которых неэффективно для минимизации функции. В связи с этим процесс построения сопряженных направлений вида (3.25) целесообразно вести циклами, начиная каждый цикл с направления наискорейшего спуска, то есть с антиградиента: как и в начальной итерации метод, запускается из текущей точки поиска по антиградиенту. Такое

обновление работы метода называется повторным стартом или *ре-стартом*. С помощью рестартов эффективность метода Флетчера – Ривса можно существенно повысить. Для этого в приведенных алгоритмах через n или $2n$ итераций переходят не к шагу 2, а к шагу 1.

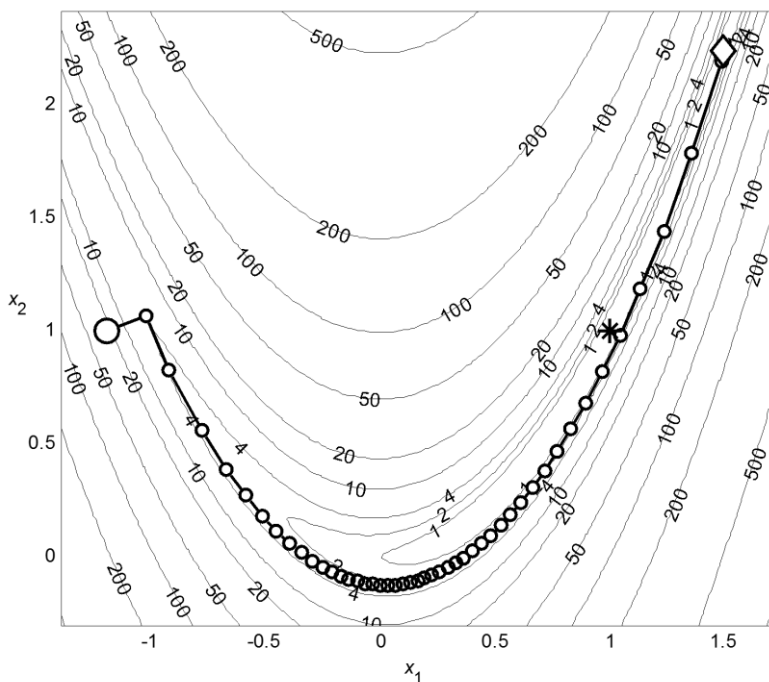


Рис. 3.6. Минимизация функции Розенброка методом Флетчера-Ривса

Пример 3.5. На рис. 3.7 представлена траектория минимизации функции Розенброка методом Флетчера – Ривса с рестартами, которые выполняются через каждые $2n$ итераций. Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод затратил 24 итерации и 327 вычислений функции.

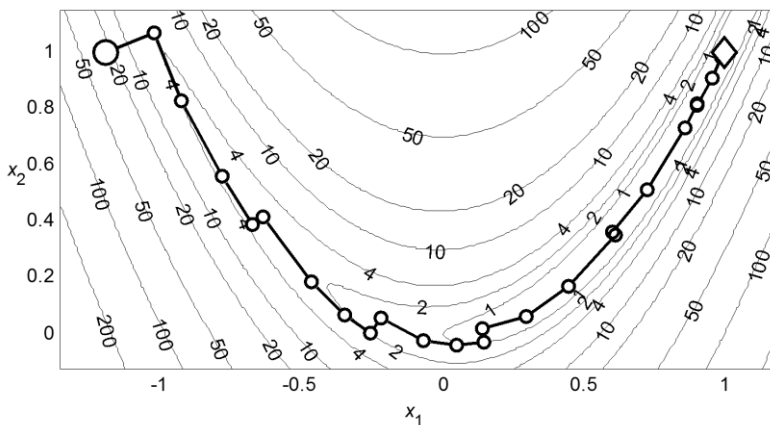


Рис. 3.7. Минимизация функции Розенброка методом Флетчера – Ривса с рестартами

3.7. Метод Полака – Рибьера

Этот метод сопряженных градиентов представлен американскими математиками Э. Полаком и Д. Рибьером в 1969 году. Для его обоснования применим первое из свойств (3.19) в виде $\mathbf{g}_k^T \mathbf{g}_{k-1} = 0$, $k = \overline{1, n-1}$. Из этой формулы следует, что

$$\mathbf{g}_k^T \mathbf{g}_k = \mathbf{g}_k^T \mathbf{g}_k - \mathbf{g}_k^T \mathbf{g}_{k-1} = \mathbf{g}_k^T (\mathbf{g}_k - \mathbf{g}_{k-1}).$$

Поэтому формулу Флетчера – Ривса (3.23) для вычисления коэффициента β_{k-1} можно представить в виде:

$$\beta_{k-1} = \frac{\mathbf{g}_k^T (\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}, \quad k = \overline{1, n-1}. \quad (3.27)$$

Эта формула называется *формулой Полака – Рибьера*, а онованный на ее использовании метод называется *методом Полака – Рибьера*. В этом методе задается начальная точка \mathbf{x}_0 , в ней вычисляется

значение градиента $\mathbf{g}_0 = \nabla f(\mathbf{x}_0)$. В направлении антиградиента производится одномерный поиск и находится следующая точка:

$$\mathbf{d}_0 = -\mathbf{g}_0, \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0). \quad (3.28)$$

Последующие итерации основаны на вычислении градиента $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$, формулах методов спуска (1.16), методов сопряженных градиентов (3.18) и формулы Полака – Рибьера (3.27):

$$\mathbf{d}_k = -\mathbf{g}_k + \beta_{k-1} \mathbf{d}_{k-1}, \quad \beta_{k-1} = \frac{\mathbf{g}_k^T (\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}, \quad k = \overline{1, n-1}, \quad (3.29)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k). \quad (3.30)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность. Формулы метода Полака – Рибьера (3.28)–(3.30) отличаются от соответствующих формул метода Флетчера – Ривса (3.24)–(3.26) только формулой вычисления коэффициента β_{k-1} . По формулам (3.28)–(3.30) составим алгоритм метода Полака – Рибьера.

Алгоритм метода Полака – Рибьера.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$ и положить $\mathbf{d} = -\mathbf{g}_x$.
2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.
3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\mathbf{g}_y = \mathbf{g}_x$.
4. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, $\beta = \mathbf{g}_x^T \cdot (\mathbf{g}_x - \mathbf{g}_y) / (\mathbf{g}_y^T \cdot \mathbf{g}_y)$.
5. Положить $\mathbf{d} = -\mathbf{g}_x + \beta \cdot \mathbf{d}$.

6. Если $\|s\| > \varepsilon$, то перейти к шагу 2.

7. Остановиться.

Этот алгоритм отличается от алгоритма метода Флетчера – Ривса только вычислением параметра β на шаге 4.

Пример 3.6. Для вычисления точки минимума квадратичной функции (1.3) с допустимой погрешностью 10^{-3} метод Полака – Рибьера затратил 3 итерации 19 вычислений функции, причем за две итерации найдено значение функции $1,8 \cdot 10^{-14}$. При этом траектория минимизации такая же, как и на рис. 3.5 для метода Флетчера – Ривса.

Пример 3.7. На рис. 3.8 представлена траектория минимизации функции Розенброка методом Полака – Рибьера. Вычисление точки минимума с допустимой погрешностью 10^{-3} потребовало 17 итераций и 266 вычислений функции.

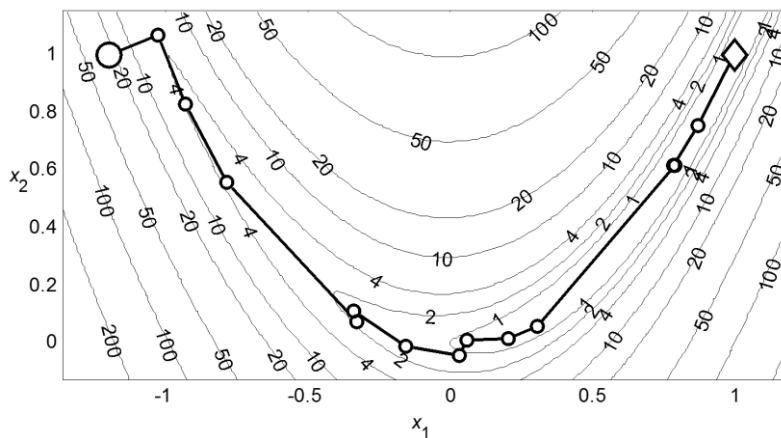


Рис. 3.8. Минимизация функции Розенброка методом Полака – Рибьера

Если все вычисления, включая одномерный поиск, проводятся с абсолютной точностью, то при минимизации квадратичной положительно определенной целевой функции методы Флетчера – Ривса и Полака – Рибьера производят одинаковые последовательности точек

поиска, то есть траектории поиска в обоих методах совпадают.

Однако при минимизации целевых функций общего вида в результате большого количества вычислительных экспериментов установлено, что метод Полака – Рибьера гораздо эффективнее метода Флетчера – Ривса. Эвристическое объяснение этого факта состоит в том, что в связи с отличием целевой функции от квадратичной функции и неточностью одномерного поиска вырабатываемые методом Полака – Рибьера направления все с меньшей точностью удовлетворяют условиям сопряженности. В результате вновь построенный по формуле (3.29) вектор направления \mathbf{d}_k становится почти ортогональным градиенту \mathbf{g}_k и метод может временно «застрять». В таком случае имеем $\mathbf{g}_{k+1} \approx \mathbf{g}_k$, поэтому по формуле Полака – Рибьера (3.27) получим значение β_k , близкое к нулю. Но тогда следующий вектор направления \mathbf{d}_{k+1} , построенный согласно (3.29), оказывается близким к $-\mathbf{g}_{k+1}$, и автоматически происходит рестарт метода. Формула Флетчера – Ривса (3.23) таким свойством не обладает, поэтому в методе Флетчера – Ривса применяют принудительные рестарты. Метод Полака – Рибьера также может применяться с рестартами.

Итак, метод Полака – Рибьера эффективен даже при наличии ошибок вычислений и обладает малой чувствительностью к ошибкам округления при проведении одномерных поисков. Ему принадлежит ведущая роль при решении задач безусловной минимизации большой размерности.

В настоящее время построено и применяется много других вариантов методов сопряженных градиентов. В частности, кроме формулы (3.27) известны формулы Хестенса – Штифеля, Диксона и Даи – Юана:

$$\beta_{k-1} = \frac{\mathbf{g}_k^T (\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{d}_{k-1}^T (\mathbf{g}_k - \mathbf{g}_{k-1})}, \quad \beta_{k-1} = -\frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{d}_{k-1}^T \mathbf{g}_{k-1}}, \quad \beta_{k-1} = \frac{\mathbf{g}_k^T \mathbf{g}_k}{\mathbf{d}_{k-1}^T (\mathbf{g}_k - \mathbf{g}_{k-1})}.$$

Все эти формулы производят одинаковые направления поиска, когда используются для минимизации квадратичной функции с положитель-

но определенной матрицей Гессе и точным одномерным поиском. Однако для нелинейной функции общего вида с неточным одномерным поиском их действия значительно отличаются.

Лабораторная работа

Тема лабораторной работы.

Разработка подпрограмм для методов сопряженных направлений.

Цель лабораторной работы.

Разработать подпрограмму метода сопряженных направлений для безусловной минимизации функций многих переменных.

Порядок выполнения лабораторной работы.

1. Получить у преподавателя вариант целевой функции $f(\mathbf{x})$ многих переменных с начальной точкой поиска минимума \mathbf{x}_0 и задание на программирование одного из методов сопряженных направлений:

- 1) метод Пауэлла;
- 2) метод Флетчера – Ривса;
- 3) метод Флетчера – Ривса с рестартами;
- 4) метод Полака – Рибьера;
- 5) метод Полака – Рибьера с рестартами.

2. Получить у преподавателя задание на программирование одного из методов одномерного поиска в многомерном пространстве:

- 1) метод дихотомии;
- 2) метод деления интервала пополам;
- 3) метод Фибоначчи;
- 4) метод золотого сечения;
- 5) метод адаптации шага;
- 6) метод квадратичной интерполяции с тремя точками;
- 7) метод квадратичной интерполяции с двумя точками;
- 8) метод секущих;
- 9) метод касательных;
- 10) метод кубической интерполяции с четырьмя точками;

11) метод кубической интерполяции с двумя точками.

3. Аналитически найти точку \mathbf{x}^* минимума заданной функции $f(\mathbf{x})$ и вычислить минимальное значение функции $f^* = f(\mathbf{x}^*)$.

4. Написать подпрограмму вычисления значений функции $f(\mathbf{x})$ с входным параметром \mathbf{x} и выходным параметром f .

5. Протестировать подпрограмму вычисления функции $f(\mathbf{x})$ при вычислении значений $f_0 = f(\mathbf{x}_0)$ и $f^* = f(\mathbf{x}^*)$, сравнить f^* с f_0 .

6. На основании алгоритмов для заданного метода сопряженных направлений и метода одномерного поиска написать компьютерные подпрограммы в соответствии с требованиями:

1) входные параметры подпрограмм: \mathbf{x}_0 – начальная точка поиска, $f(\mathbf{x})$ – целевая функция, ε – допустимая погрешность;

2) выходные параметры подпрограмм: \mathbf{X} – массив всех точек поиска \mathbf{x}_k ; \mathbf{F} – массив значений функции $f_k = f(\mathbf{x}_k)$; \mathbf{x} и $f_{\mathbf{x}}$ – лучшая точка поиска и значение в ней функции;

3) на каждой итерации выводить на экран монитора строку, содержащую номер точки поиска k , шаг перехода в новую точку $h_k = \|\mathbf{s}_k\| = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|$, значение функции f_k и значение вектора переменных \mathbf{x}_k ;

4) итерации продолжаются до тех пор, пока выполняется неравенство $h_k > \varepsilon$;

5) после окончания процесса оптимизации на экране должна отображаться таблица, имеющая соответствующий заголовок и представляющая процесс минимизации функции;

6) под таблицей необходимо отобразить количество вычислений целевой функции, конечный шаг поиска, минимальное значение функции и соответствующую ему точку поиска.

7. Написать общую программу для выполнения вычислительного процесса минимизации целевой функции многих переменных и отображения на экране этого процесса в соответствии с требованиями:

1) за основу общей программы для минимизации функции взять общую программу из предыдущей лабораторной работы и модифицировать эту программу;

2) задать в программе значения \mathbf{x}_0 , h , ε ;

3) путем вызова разработанной подпрограммы заданного метода сопряженных направлений с использованием подпрограмм вычисления функции $f(\mathbf{x})$ и заданного метода одномерного поиска, а также значений \mathbf{x}_0 , h , ε реализовать итерационный процесс минимизации функции с построением таблицы процесса на экране и получением массивов значений вектора переменных \mathbf{X} и функции \mathbf{F} , лучшей точки поиска \mathbf{x} и значения в ней функции $f_{\mathbf{x}}$;

4) с помощью подпрограммы графического отображения итерационного процесса по массивам \mathbf{X} и \mathbf{F} отобразить графики процесса минимизации целевой функции многих переменных на экране.

8. Выполнить анализ полученных результатов:

1) на основании табличной и графической информации о работе метода одномерного поиска в многомерном пространстве определить конечные точки метода Свенна и показать найденный этим методом интервал неопределенности;

2) на нескольких начальных итерациях по табличным данным и графикам сопоставить начало работы метода сопряженных направлений с его рабочими формулами;

3) оценить эффективность метода сопряженных направлений и сопоставить ее с эффективностью методов многомерной безусловной оптимизации из предыдущих лабораторных работ.

9. Ответить на вопросы преподавателя по теме лабораторной работы.

10. Оформить и сдать отчет о проведенной лабораторной работе.

С о д е р ж а н и е о т ч е т а :

1) титульный лист установленного образца с указанием организации, названия учебной дисциплины, темы работы, номера варианта, исполнителя и принимающего, города, года;

- 2) постановка задачи минимизации целевой функции, включающая задание функции, начальную точку и точку минимума;
- 3) теоретическое описание метода сопряженных направлений для безусловной минимизации функций многих переменных;
- 4) алгоритм метода сопряженных направлений;
- 5) теоретическое описание метода одномерного поиска в многомерном пространстве;
- 6) алгоритм метода одномерного поиска;
- 7) распечатки используемых компьютерных программ;
- 8) табличное представление процесса минимизации и его итоги – количество вычислений значений целевой функции, минимальное значение функции и соответствующее ему значение точки минимума, достигнутую точность;
- 9) трехмерный график целевой функции и траектории поиска со всеми точками;
- 10) двумерный график линий уровня функции и траектории поиска со всеми точками;
- 11) анализ полученных результатов;
- 12) выводы о проделанной лабораторной работе.

Контрольные вопросы

1. Представьте в общем виде положительно определенную квадратичную функцию.
2. Запишите формулу для градиента квадратичной функции.
3. Приведите свойства квадратичной функции.
4. Найдите минимум положительно определенной квадратичной функции.
5. Выведите условие точного одномерного поиска для функции многих переменных.
6. Выведите условие точного одномерного поиска для квадратичной функции многих переменных.
7. Дайте определение сопряженных направлений.

8. Интерпретируйте понятие сопряженных векторов для единичной матрицы.

9. Сформулируйте и докажите лемму о линейной независимости сопряженных векторов.

10. Сформулируйте следствие леммы о линейной независимости сопряженных векторов.

11. Сформулируйте и докажите теорему о свойстве параллельного подпространства.

12. Сформулируйте и докажите основную теорему методов сопряженных направлений.

13. Раскройте принцип расширяющихся подпространств и дайте его геометрическую интерпретацию.

14. Какие общие свойства методов сопряженных направлений вытекают из основной теоремы методов сопряженных направлений.

15. В чем заключается основная идея метода Пауэлла?

16. Запишите итерационные формулы метода Пауэлла.

15. Опишите метод Пауэлла.

16. По какому принципу задается направление шага в методе Пауэлла?

17. Как вычисляется величина шага в методе Пауэлла?

18. Как заканчиваются вычисления в методе Пауэлла?

19. Какого порядка метод Пауэлла?

20. Составьте алгоритм метода Пауэлла.

21. Укажите достоинства и недостатки метода Пауэлла.

22. Дайте определение методов сопряженных градиентов.

23. Сформулируйте и докажите основную теорему методов сопряженных градиентов.

24. Какие общие свойства методов сопряженных градиентов вытекают из основной теоремы этих методов.

25. Запишите формулу Флетчера – Ривса.

26. Запишите итерационные формулы метода Флетчера – Ривса.

27. Опишите метод Флетчера – Ривса.

28. Как вычисляется величина шага в методе Флетчера – Ривса?

29. Как заканчиваются вычисления в методе Флетчера – Ривса?
30. Какого порядка метод Флетчера – Ривса?
31. Составьте алгоритм метода Флетчера – Ривса.
32. Составьте алгоритм экономного метода Флетчера – Ривса.
33. Укажите достоинства и недостатки метода Флетчера – Ривса.
34. Дайте определение рестарта в методе Флетчера – Ривса.
35. Обоснуйте необходимость применения рестартов в методе Флетчера – Ривса.
36. Раскройте роль рестартов в повышении эффективности метода Флетчера – Ривса.
37. Запишите и обоснуйте формулу Полака – Рибьера.
38. Запишите итерационные формулы метода Полака – Рибьера.
39. Опишите метод Полака – Рибьера.
40. Как вычисляется величина шага в методе Полака – Рибьера?
41. Как заканчиваются вычисления в методе Полака – Рибьера?
42. Какого порядка метод Полака – Рибьера?
43. Составьте алгоритм метода Полака – Рибьера.
44. Чем отличается алгоритм метода Полака – Рибьера от алгоритма метода Флетчера – Ривса?
45. Укажите достоинства и недостатки метода Полака – Рибьера.
46. Обоснуйте преимущество метода Полака – Рибьера перед методом Флетчера – Ривса.
47. Какие вы еще знаете методы сопряженных градиентов?
48. Запишите и обоснуйте формулу Хестенса – Штифеля.
49. Запишите и обоснуйте формулу Диксона.
50. Запишите и обоснуйте формулу Дай – Юана.

4. КВАЗИНЬЮТОНОВСКИЕ МЕТОДЫ

Не существует единственного самого эффективного метода для безусловной минимизации функции многих переменных. В этом разделе рассматриваются квазиньютоновские методы первого порядка, основанные на формуле Ньютона и на аппроксимации матрицы Гессе или обратной к ней матрицы с использованием значений градиента. Приводятся теоретические основы квазиньютоновских методов, вводится понятие поправки, уточняющей аппроксимацию и вычисляемой по градиентам. Обосновывается метод Бройдена с поправкой ранга один, анализируются его свойства. Вводится поправка ранга два, конструируется метод Девидона – Флетчера – Пауэлла с аппроксимацией обратной матрицы Гессе, рассматриваются его свойства, перечисляются преимущества и недостатки. Обосновывается метод Бройдена – Флетчера – Гольдфарба – Шанно с аппроксимацией матрицы Гессе, приводящей к необходимости решения системы уравнений, и отмечается его малая чувствительность к точности одномерного поиска. Модифицируется метод Бройдена – Флетчера – Гольдфарба – Шанно путем перехода к аппроксимации обратной матрицы Гессе. Приводятся алгоритмы для всех рассмотренных методов. Рассматривается скорость сходимости методов многомерной безусловной минимизации и проводится сравнение методов. Дается описание лабораторной работы по изучению квазиньютоновских методов оптимизации.

4.1. Основы квазиньютоновских методов

Модифицированный метод Ньютона с одномерным поиском (2.5) для минимизации целевой функции $f(\mathbf{x})$ с $\mathbf{x} \in R^n$ представляется итерационной формулой

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k, \quad (4.1)$$

где λ_k – значение параметра одномерной минимизации целевой функции из точки \mathbf{x}_k в направлении $\mathbf{d}_k = -\mathbf{H}_k^{-1} \mathbf{g}_k$, $\mathbf{H}_k = \nabla^2 f(\mathbf{x}_k)$ – матрица Гессе, $\mathbf{g}_k = \nabla f(\mathbf{x}_k)$ – градиент. Модифицированный метод Ньютона – это метод второго порядка. Для многих задач оптимизации вычисление матрицы Гессе, состоящей из вторых частных производных, требует больших затрат машинного времени.

Квазиньютоновские методы также основаны на итерационной формуле (4.1). Однако при вычислении направления одномерного поиска \mathbf{d}_k выполняется аппроксимация матрицы Гессе или обратной к ней матрицы с использованием значений градиента. Таким образом, все квазиньютоновские методы – это методы первого порядка.

Рассмотрим принципы построения аппроксимирующей матрицы для обратной матрицы Гессе $\mathbf{G}(\mathbf{x}) = \mathbf{H}(\mathbf{x})^{-1}$. Поскольку матрицы $\mathbf{H}(\mathbf{x})$ и $\mathbf{G}(\mathbf{x})$ симметрические, то начальное приближение матрицы $\mathbf{G}(\mathbf{x})$ задается в виде некоторой симметрической положительно определенной матрицы \mathbf{G}_0 .

Положительная определенность матрицы обеспечивает соответствующее направление одномерного поиска $\mathbf{d}_0 = -\mathbf{G}_0 \mathbf{g}_0$ из начальной точки \mathbf{x}_0 с градиентом целевой функции \mathbf{g}_0 как направление спуска. Обычно полагают $\mathbf{G}_0 = \mathbf{E}$, тогда $\mathbf{d}_0 = -\mathbf{g}_0$ – направление наискорейшего спуска. Из начальной точки \mathbf{x}_0 в направлении \mathbf{d}_0 выполняют одномерный поиск минимума целевой функции:

$$\lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0), \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0.$$

В точке \mathbf{x}_1 вычисляется значение градиента целевой функции \mathbf{g}_1 . Для последующих итераций построение аппроксимирующей матрицы основано на свойстве (3.8) квадратичной функции (3.1) с $\mathbf{A} = \mathbf{H}$

$$\mathbf{g}_{k+1} - \mathbf{g}_k = \mathbf{H}(\mathbf{x}_{k+1} - \mathbf{x}_k). \quad (4.2)$$

С обозначениями

$$\mathbf{p}_k = \mathbf{g}_{k+1} - \mathbf{g}_k, \quad \mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \quad (4.3)$$

свойство (4.2) примет вид

$$\mathbf{p}_k = \mathbf{H}\mathbf{s}_k. \quad (4.4)$$

Умножая это равенство слева на обратную матрицу Гессе $\mathbf{G} = \mathbf{H}^{-1}$, получим $\mathbf{G}\mathbf{p}_k = \mathbf{s}_k$. Если известна аппроксимация матрицы Гессе \mathbf{G}_k в точке \mathbf{x}_k , то следующее ее приближение \mathbf{G}_{k+1} в точке \mathbf{x}_{k+1} должно удовлетворять уравнению

$$\mathbf{G}_{k+1}\mathbf{p}_k = \mathbf{s}_k. \quad (4.5)$$

Это матрично-векторное уравнение называется *квазиньютоновским условием*. Оно представляет собой систему n линейных алгебраических уравнений. С учетом свойства симметричности матрица \mathbf{G}_{k+1} имеет

$$N = n + (n^2 - n)/2 = n(n+1)/2$$

неизвестных элементов, то есть система линейных алгебраических уравнений (4.5) содержит больше неизвестных, чем уравнений. Для ее решения необходимо дополнительное условие, которое записывается в виде *уравнения коррекции*

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \Delta\mathbf{G}_k. \quad (4.6)$$

Здесь $\Delta\mathbf{G}_k$ – симметрическая матрица, называемая *поправкой* и формируемая так, чтобы выполнялась система уравнений (4.5). Различные квазиньютоновские методы отличаются между собой формулами для поправки $\Delta\mathbf{G}_k$.

После определения аппроксимирующей матрицы \mathbf{G}_k на последующих итерациях квазиньютоновского метода направление одномерного поиска определяется по формуле

$$\mathbf{d}_k = -\mathbf{G}_k \mathbf{g}_k. \quad (4.7)$$

В этом направлении из точки \mathbf{x}_k выполняется одномерный поиск:

$$\lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k. \quad (4.8)$$

Итерации продолжаются, пока не выполнится условие окончания процесса оптимизации

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| \leq \varepsilon,$$

где ε – допустимая погрешность.

Направление поиска (4.7) отличается от направления метода наискорейшего спуска в формуле (1.20) $-\mathbf{g}_k$ умножением $-\mathbf{g}_k$ на изменяющуюся на каждой итерации матрицу \mathbf{G}_k . Направление (4.7) можно представить как направление метода наискорейшего спуска в пространстве переменных с измененной метрикой. Поэтому квазиньютоновские методы называются также *методами переменной метрики*.

4.2. Метод Бroyдена

Для того, чтобы аппроксимация обратной матрицы Гессе давала симметрическую матрицу, поправка $\Delta \mathbf{G}_k$ в формуле (4.6) должна быть также симметрической матрицей. Возьмем произвольный ненулевой вектор $\mathbf{u} = (u_1 \ u_2 \ \dots \ u_n)^T$ и построим матрицу

$$\mathbf{u}\mathbf{u}^T = \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ u_n \end{pmatrix} (u_1 \ u_2 \ \dots \ u_n) = \begin{pmatrix} u_1 u_1 & u_1 u_2 & \dots & u_1 u_n \\ u_2 u_1 & u_2 u_2 & \dots & u_2 u_n \\ \dots & \dots & \dots & \dots \\ u_n u_1 & u_n u_2 & \dots & u_n u_n \end{pmatrix}.$$

Это симметрическая матрица с пропорциональными строками, поэтому она имеет единичный ранг $\text{rank}(\mathbf{u}\mathbf{u}^T) = 1$. Сформируем поправку

ранга один в виде $\Delta \mathbf{G}_k = \alpha \mathbf{u} \mathbf{u}^T$, где α – некоторый вещественный коэффициент. По формуле (4.6)

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T. \quad (4.9)$$

Система линейных алгебраических уравнений (4.5) принимает вид

$$(\mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T) \mathbf{p}_k = \mathbf{s}_k.$$

Раскрывая скобки, имеем

$$\mathbf{G}_k \mathbf{p}_k + \alpha \mathbf{u} \mathbf{u}^T \mathbf{p}_k = \mathbf{s}_k,$$

то есть

$$(\alpha \mathbf{u}^T \mathbf{p}_k) \mathbf{u} = \mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k,$$

где скобки содержат скалярную величину. Это векторное равенство удовлетворяется, если положить $\mathbf{u} = \mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k$ и $\alpha = 1/(\mathbf{u}^T \mathbf{p}_k)$ при условии $\mathbf{u}^T \mathbf{p}_k \neq 0$. Тогда равенство (4.9) примет вид

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \frac{(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T}{(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k}. \quad (4.10)$$

Эта формула коррекции аппроксимации обратной матрицы Гессе представлена в 1967 году английским математиком Ч. Д. Бройденом и называется *формулой Бройдена*, а использующий ее квазиньютоновский метод называется *методом Бройдена*. Таким образом, метод Бройдена основан на формулах (4.3), (4.7), (4.8) и (4.10). Первая итерация начинается из заданной начальной точки \mathbf{x}_0 , и в направлении антиградиента выполняется одномерный поиск:

$$\mathbf{G}_0 = \mathbf{E}, \quad \mathbf{d}_0 = -\mathbf{g}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0), \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0. \quad (4.11)$$

Последующие итерации для $k = 1, 2, 3, \dots$ выполняются по формулам:

$$\mathbf{p}_{k-1} = \mathbf{g}_k - \mathbf{g}_{k-1}, \quad \mathbf{s}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}, \quad (4.12)$$

$$\mathbf{u}_{k-1} = \mathbf{s}_{k-1} - \mathbf{G}_{k-1} \mathbf{p}_{k-1}, \quad \mathbf{G}_k = \mathbf{G}_{k-1} + \frac{\mathbf{u}_{k-1} \mathbf{u}_{k-1}^T}{\mathbf{u}_{k-1}^T \mathbf{p}_{k-1}}, \quad (4.13)$$

$$\mathbf{d}_k = -\mathbf{G}_k \mathbf{g}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k. \quad (4.14)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon,$$

где ε – допустимая погрешность. По формулам (4.11)–(4.14) составим алгоритм метода Бroyдена.

Алгоритм метода Бройдена.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$ и положить $\mathbf{d} = -\mathbf{g}_x$, $\mathbf{G} = \mathbf{E}$.

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\mathbf{g}_y = \mathbf{g}_x$.

4. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, $\mathbf{p} = \mathbf{g}_x - \mathbf{g}_y$, $\mathbf{u} = \mathbf{s} - \mathbf{G} \cdot \mathbf{p}$.

5. Положить $\mathbf{G} = \mathbf{G} + \mathbf{u} \cdot \mathbf{u}^T / (\mathbf{u}^T \cdot \mathbf{p})$.

6. Вычислить $\mathbf{d} = -\mathbf{G} \cdot \mathbf{g}_x$.

7. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 2.

8. Остановиться.

Пример 4.1. На рис. 4.1 представлена траектория минимизации квадратичной функции (1.3) методом Бройдена. Для вычисления точки минимума с допустимой погрешностью 10^{-3} метод затратил 3 итерации и 19 вычислений функции. За две итерации получено приближение точки минимума с погрешностью $9,7 \cdot 10^{-8}$. При этом траектория минимизации квадратичной функции методом Бройдена такая же, как и траектория минимизации этой же функции методом Флетчера – Ривса из примера 3.3, представленная на рис. 3.5.

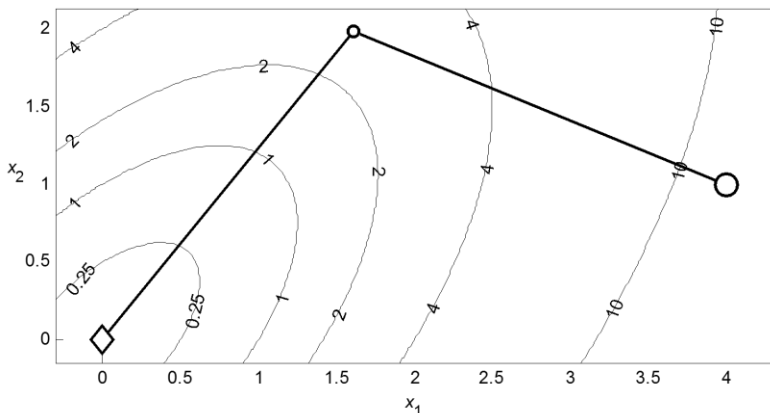


Рис. 4.1. Минимизация квадратичной функции методом Бройдена

Пример 4.2. На рис. 4.2 представлена траектория минимизации функции Розенброка методом Бройдена. Вычисление точки минимума с допустимой погрешностью 10^{-3} потребовало 17 итераций и 269 вычислений функции, что сравнимо с результатами метода Полака – Рибьера из примера 3.7. Траектория поиска такая же, как и на рис. 3.8 для метода Полака – Рибьера.

4.3. Свойства метода Бройдена

Рассмотрим и обоснуем свойства метода Бройдена.

Свойство 1. Если в формуле Бройдена (4.10) матрица \mathbf{G}_k симметрическая, то матрица \mathbf{G}_{k+1} также симметрическая.

Доказательство. Пусть матрица \mathbf{G}_k симметрическая, то есть $\mathbf{G}_k^T = \mathbf{G}_k$. Тогда по формуле Бройдена (4.10) с обозначениями $\mathbf{u} = \mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k$ и $\alpha = 1/(\mathbf{u}^T \mathbf{p}_k)$ придем к формуле (4.9). Применяя свойства матриц, получим:

$$\mathbf{G}_{k+1}^T = (\mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T)^T = \mathbf{G}_k^T + \alpha (\mathbf{u}^T)^T \mathbf{u}^T = \mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T.$$

Следовательно, $\mathbf{G}_{k+1}^T = \mathbf{G}_{k+1}$. \square

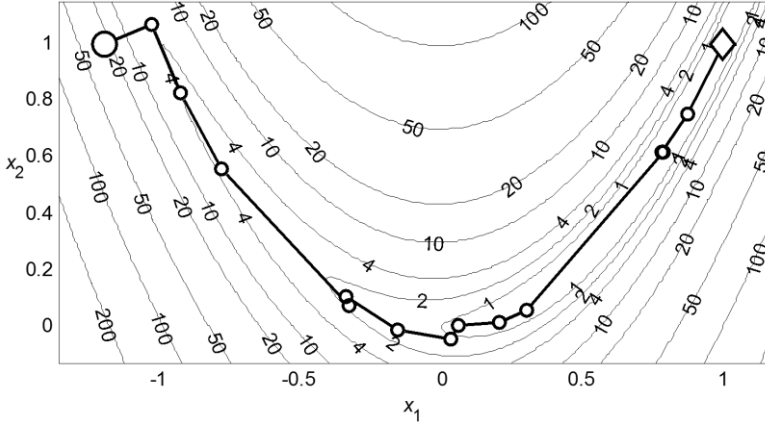


Рис. 4.2. Минимизация функции Розенброка методом Бройдена

Свойство 2. Если в формуле Бройдена (4.10) матрица \mathbf{G}_k положительно определенная и $(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k > 0$, то матрица \mathbf{G}_{k+1} также положительно определенная.

Доказательство. Пусть матрица \mathbf{G}_k положительно определенная, то есть $\mathbf{x}^T \mathbf{G}_k \mathbf{x} > 0 \quad \forall \mathbf{x} \neq \mathbf{0}$. Тогда по формуле Бройдена (4.10) с обозначениями $\mathbf{u} = \mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k$ и $\alpha = 1/(\mathbf{u}^T \mathbf{p}_k)$ получим:

$$\mathbf{x}^T \mathbf{G}_{k+1} \mathbf{x} = \mathbf{x}^T \mathbf{G}_k \mathbf{x} + \alpha \mathbf{x}^T \mathbf{u} \mathbf{u}^T \mathbf{x} = \mathbf{x}^T \mathbf{G}_k \mathbf{x} + \alpha (\mathbf{x}^T \mathbf{u})^2.$$

При $\mathbf{x} \neq \mathbf{0}$ в правой части этого равенства первое слагаемое положительно, а второе неотрицательно, если $\alpha = 1/[(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k] > 0$, то есть если $(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k > 0$. Тогда $\mathbf{x}^T \mathbf{G}_{k+1} \mathbf{x} > 0 \quad \forall \mathbf{x} \neq \mathbf{0}$. \square

Свойство 3. При минимизации квадратичной функции с по-

ложительно определенной матрицей Гессе \mathbf{H} методом Бroyдена выполняются равенства:

$$\mathbf{G}_k \mathbf{p}_i = \mathbf{s}_i, \quad i = \overline{0, k-1}. \quad (4.15)$$

Доказательство. Применим метод математической индукции. При $k=1$ равенство (4.15) вытекает из квазиньютоновского условия (4.5) в виде $\mathbf{G}_1 \mathbf{p}_0 = \mathbf{s}_0$. Предположим, что равенства (4.15) выполнены для некоторого k . Тогда для $i = \overline{0, k-1}$ с учетом свойства квадратичной функции (4.4) в виде $\mathbf{p}_i = \mathbf{H} \mathbf{s}_i$ получим:

$$(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_i = \mathbf{s}_k^T \mathbf{p}_i - \mathbf{p}_k^T \mathbf{G}_k \mathbf{p}_i = \mathbf{s}_k^T \mathbf{p}_i - \mathbf{p}_k^T \mathbf{s}_i = \mathbf{s}_k^T \mathbf{H} \mathbf{s}_i - \mathbf{s}_k^T \mathbf{H} \mathbf{s}_i = 0.$$

Отсюда по формуле Бroyдена (4.10) и гипотезе индукции имеем:

$$\mathbf{G}_{k+1} \mathbf{p}_i = \mathbf{G}_k \mathbf{p}_i + \frac{(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_i}{(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k} = \mathbf{G}_k \mathbf{p}_i = \mathbf{s}_i, \quad i = \overline{0, k-1}.$$

С учетом квазиньютоновского условия (4.5) $\mathbf{G}_{k+1} \mathbf{p}_k = \mathbf{s}_k$ окончательно получим: $\mathbf{G}_{k+1} \mathbf{p}_i = \mathbf{s}_i, \quad i = \overline{0, k}$. \square

Свойство 4. Если при минимизации квадратичной функции с положительно определенной матрицей Гессе \mathbf{H} метод Бroyдена производит линейно независимые направления $\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{n-1}$, то $\mathbf{G}_n = \mathbf{H}^{-1}$ и минимум находится не более чем за $n+1$ итерацию.

Доказательство. При сделанных предположениях после n итераций метода Бroyдена выполняются равенства (4.15) при $k=n$ в виде $\mathbf{G}_n \mathbf{p}_i = \mathbf{s}_i, \quad i = \overline{0, n-1}$. Отсюда с учетом свойства квадратичной функции (4.4) в виде $\mathbf{p}_i = \mathbf{H} \mathbf{s}_i$ имеем: $\mathbf{G}_n \mathbf{H} \mathbf{s}_i = \mathbf{s}_i, \quad i = \overline{0, n-1}$. Поскольку векторы \mathbf{s}_i образуют n линейно независимых направлений, которые можно представить столбцами невырожденной матрицы \mathbf{S} , то справедливо матричное уравнение $\mathbf{G}_n \mathbf{H} \mathbf{S} = \mathbf{S}$. Умножая это равенство

справа на матрицу \mathbf{S}^{-1} , получим $\mathbf{G}_n \mathbf{H} = \mathbf{E}$, откуда $\mathbf{G}_n = \mathbf{H}^{-1}$. Поэтому следующая итерация метода Бройдена является итерацией метода Ньютона и приведет к точке минимума квадратичной функции. \square

Заметим, что в доказательстве этого свойства не используется условие точного одномерного поиска.

Свойство 5. *Если при минимизации квадратичной функции с положительно определенной матрицей Гессе \mathbf{H} методом Бройдена выполняется точный одномерный поиск, то направления поиска $\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{n-1}$ являются \mathbf{H} -сопряженными.*

Доказательство. При выполнении условий данного утверждения требуется доказать, что

$$\mathbf{s}_i^T \mathbf{H} \mathbf{s}_k = 0, \quad i = \overline{0, k-1}. \quad (4.16)$$

Докажем это свойство методом математической индукции. При $k=1$ с учетом (4.14) имеем:

$$\mathbf{s}_0^T \mathbf{H} \mathbf{s}_1 = \mathbf{s}_0^T \mathbf{H} \lambda_1 \mathbf{d}_1 = \mathbf{s}_0^T \mathbf{H} (-\lambda_1 \mathbf{G}_1 \mathbf{g}_1) = -\lambda_1 \mathbf{s}_0^T \mathbf{H} \mathbf{G}_1 \mathbf{g}_1 = -\lambda_1 (\mathbf{G}_1 \mathbf{H} \mathbf{s}_0)^T \mathbf{g}_1.$$

Отсюда по свойству 3 в виде $\mathbf{G}_1 \mathbf{p}_0 = \mathbf{s}_0$ и по условию точного одномерного поиска (3.7) в виде $\mathbf{s}_0^T \mathbf{g}_1 = 0$ получим:

$$\mathbf{s}_0^T \mathbf{H} \mathbf{s}_1 = -\lambda_1 (\mathbf{G}_1 \mathbf{p}_0)^T \mathbf{g}_1 = -\lambda_1 \mathbf{s}_0^T \mathbf{g}_1 = 0.$$

Предположим, что равенства (4.16) выполнены для некоторого k . Докажем, что они выполняются и для $k+1$. Для $i = \overline{0, k}$ имеем:

$$\begin{aligned} \mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} &= \mathbf{s}_i^T \mathbf{H} \lambda_{k+1} \mathbf{d}_{k+1} = \lambda_{k+1} \mathbf{s}_i^T \mathbf{H} (-\mathbf{G}_{k+1} \mathbf{g}_{k+1}) = -\lambda_{k+1} \mathbf{s}_i^T \mathbf{H} \mathbf{G}_{k+1} \mathbf{g}_{k+1} = \\ &= -\lambda_{k+1} (\mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i)^T \mathbf{g}_{k+1} = -\lambda_{k+1} (\mathbf{G}_{k+1} \mathbf{p}_i)^T \mathbf{g}_{k+1}. \end{aligned}$$

По свойству 3 получим

$$\mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} = -\lambda_{k+1} \mathbf{s}_i^T \mathbf{g}_{k+1}, \quad i = \overline{0, k}. \quad (4.17)$$

Очевидно, что

$$\mathbf{g}_{k+1} = \mathbf{g}_{i+1} + (\mathbf{g}_{i+2} - \mathbf{g}_{i+1}) + (\mathbf{g}_{i+3} - \mathbf{g}_{i+2}) + \dots + (\mathbf{g}_{k+1} - \mathbf{g}_k).$$

С учетом обозначений (4.3) имеем:

$$\mathbf{g}_{k+1} = \mathbf{g}_{i+1} + \sum_{j=i+1}^k (\mathbf{g}_{j+1} - \mathbf{g}_j) = \mathbf{g}_{i+1} + \sum_{j=i+1}^k \mathbf{p}_j.$$

По свойству квадратичной функции (4.4) в виде $\mathbf{p}_j = \mathbf{H} \mathbf{s}_j$ получим

$$\mathbf{g}_{k+1} = \mathbf{g}_{i+1} + \sum_{j=i+1}^k \mathbf{H} \mathbf{s}_j. \quad (4.18)$$

Равенство (4.17) примет вид

$$\mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} = -\lambda_{k+1} \mathbf{s}_i^T \left(\mathbf{g}_{i+1} + \sum_{j=i+1}^k \mathbf{H} \mathbf{s}_j \right) = -\lambda_{k+1} \left(\mathbf{s}_i^T \mathbf{g}_{i+1} + \sum_{j=i+1}^k \mathbf{s}_i^T \mathbf{H} \mathbf{s}_j \right).$$

По условию точного одномерного поиска (3.7) в виде $\mathbf{s}_i^T \mathbf{g}_{i+1} = 0$ и сделанного предположения индукции (4.16) окончательно получим для $i = \overline{0, k}$ условие сопряженности $\mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} = 0$. Следовательно, равенства (4.16) выполняются для всех $k = \overline{1, n-1}$. \square

Это свойство показывает, что метод Бройдена является методом сопряженных направлений, поэтому он минимизирует квадратичную функцию с положительно определенной матрицей Гессе \mathbf{H} при выполнении точного одномерного поиска не более чем за n итераций.

В случае, когда целевая функция не является квадратичной, применение уравнения (4.10) может привести к нежелательным явлениям. Во-первых, матрица \mathbf{G}_k может перестать быть положительно определенной. Во-вторых, поправка $\Delta \mathbf{G}_k$ может стать неограниченной. В-третьих, если направление \mathbf{s}_k случайно совпадет с направлением

предыдущей итерации, матрица \mathbf{G}_{k+1} становится вырожденной или неопределенной. В алгоритме Бroyдена это тоже будет иметь место, если либо $\mathbf{G}_k \mathbf{p}_k = \mathbf{s}_k$, либо $(\mathbf{s}_k - \mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k = 0$. Тогда знаменатель формулы (4.10) обращается в нуль. Эта особенность снижает надежность метода Бroyдена.

4.4. Метод Девидона – Флетчера – Пауэлла

По аналогии с формулой (4.9) коррекции ранга один рассмотрим формулу коррекции ранга два

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T + \beta \mathbf{v} \mathbf{v}^T, \quad (4.19)$$

где используются ненулевые векторы $\mathbf{u}, \mathbf{v} \in R^n$ и скаляры $\alpha, \beta \in R$. Для системы линейных алгебраических уравнений (4.5) получим

$$(\mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T + \beta \mathbf{v} \mathbf{v}^T) \mathbf{p}_k = \mathbf{s}_k.$$

Раскрывая скобки, после преобразований имеем

$$\mathbf{G}_k \mathbf{p}_k + (\alpha \mathbf{u}^T \mathbf{p}_k) \mathbf{u} + (\beta \mathbf{v}^T \mathbf{p}_k) \mathbf{v} = \mathbf{s}_k,$$

где скобки содержат скалярные величины. Это уравнение превратится в тождество, если положить $\mathbf{u} = \mathbf{s}_k$, $\alpha = 1/(\mathbf{u}^T \mathbf{p}_k)$, $\mathbf{v} = \mathbf{G}_k \mathbf{p}_k$, $\beta = -1/(\mathbf{v}^T \mathbf{p}_k)$. Тогда равенство (4.19) примет вид

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{p}_k} - \frac{\mathbf{G}_k \mathbf{p}_k (\mathbf{G}_k \mathbf{p}_k)^T}{(\mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k}. \quad (4.20)$$

Эта формула называется *формулой Девидона – Флетчера – Пауэлла* (ДФП), а использующий ее квазиньютоновский метод называется *методом Девидона – Флетчера – Пауэлла*. Формула (4.14) впервые представлена в 1959 году американским математиком В. Девидоном как часть метода оптимизации, а в 1963 году метод, основанный на фор-

муде (4.20), развит английскими математиками Р. Флетчером и М. Д. Д. Пауэллом. Таким образом, метод ДФП основан на формулах (4.3), (4.7), (4.8) и (4.20). Как и в методе Бройдена первая итерация начинается из заданной начальной точки \mathbf{x}_0 , и в направлении антиградиента выполняется одномерный поиск:

$$\mathbf{G}_0 = \mathbf{E}, \quad \mathbf{d}_0 = -\mathbf{g}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0), \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0. \quad (4.21)$$

Последующие итерации для $k=1, 2, 3, \dots$ выполняются по формулам:

$$\mathbf{p}_{k-1} = \mathbf{g}_k - \mathbf{g}_{k-1}, \quad \mathbf{s}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}, \quad (4.22)$$

$$\mathbf{v}_{k-1} = \mathbf{G}_{k-1} \mathbf{p}_{k-1}, \quad \mathbf{G}_k = \mathbf{G}_{k-1} + \frac{\mathbf{s}_{k-1} \mathbf{s}_{k-1}^T}{\mathbf{s}_{k-1}^T \mathbf{p}_{k-1}} - \frac{\mathbf{v}_{k-1} \mathbf{v}_{k-1}^T}{\mathbf{v}_{k-1}^T \mathbf{p}_{k-1}}, \quad (4.23)$$

$$\mathbf{d}_k = -\mathbf{G}_k \mathbf{g}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k. \quad (4.24)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon.$$

По формулам (4.21)–(4.24) составим алгоритм метода ДФП.

Алгоритм метода ДФП.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$ и положить $\mathbf{d} = -\mathbf{g}_x$, $\mathbf{G} = \mathbf{E}$.

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\mathbf{g}_y = \mathbf{g}_x$.

4. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, $\mathbf{p} = \mathbf{g}_x - \mathbf{g}_y$, $\mathbf{v} = \mathbf{G} \cdot \mathbf{p}$.

5. Положить $\mathbf{G} = \mathbf{G} + \mathbf{s} \cdot \mathbf{s}^T / (\mathbf{s}^T \cdot \mathbf{p}) - \mathbf{v} \cdot \mathbf{v}^T / (\mathbf{v}^T \cdot \mathbf{p})$.

6. Вычислить $\mathbf{d} = -\mathbf{G} \cdot \mathbf{g}_x$.

7. Если $\|s\| > \varepsilon$, то перейти к шагу 2.

8. Остановиться.

Пример 4.3. Для вычисления точки минимума квадратичной функции (1.3) с допустимой погрешностью 10^{-3} метод Девидона – Флетчера – Пауэлла затратил 3 итерации и 19 вычислений функции. При этом траектория минимизации квадратичной функции методом ДФП такая же, как и траектория минимизации этой же функции методом Бройдена из примера 4.1, представленная на рис. 4.1.

Пример 4.4. На рис. 4.3 представлена траектория минимизации функции Розенброка методом Девидона – Флетчера – Пауэлла. Одномерный поиск с начальным единичным шагом производился комбинацией метода Свенна и метода квадратичной интерполяции с тремя точками при допустимой погрешности шага 10^{-5} . Для нахождения точки минимума с допустимой погрешностью 10^{-3} метод ДФП использовал 17 итераций и 265 вычислений функции. Траектория минимизации такая же, как и траектория метода Бройдена из примера 4.2, представленная на рис. 4.2.

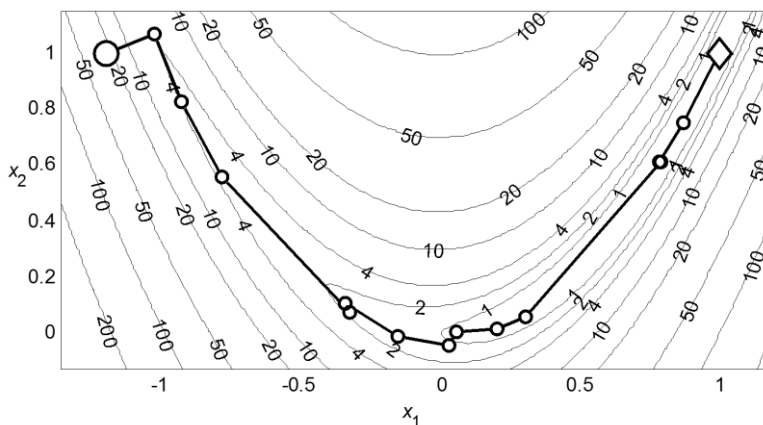


Рис. 4.3. Минимизация функции Розенброка методом ДФП

4.5. Свойства метода Девидона – Флетчера – Пауэлла

Эффективность метода ДФП определяется его свойствами.

Свойство 1. Если в формуле ДФП (4.20) матрица \mathbf{G}_k симметрическая, то матрица \mathbf{G}_{k+1} также симметрическая.

Доказательство. Пусть матрица \mathbf{G}_k симметрическая, то есть $\mathbf{G}_k^T = \mathbf{G}_k$. Тогда, по формуле ДФП (4.20) с обозначениями $\mathbf{u} = \mathbf{s}_k$, $\alpha = 1/(\mathbf{u}^T \mathbf{p}_k)$, $\mathbf{v} = \mathbf{G}_k \mathbf{p}_k$, $\beta = -1/(\mathbf{v}^T \mathbf{p}_k)$ придем к формуле (4.19). Используя свойства матриц, получим:

$$\begin{aligned}\mathbf{G}_{k+1}^T &= (\mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T + \beta \mathbf{v} \mathbf{v}^T)^T = \mathbf{G}_k^T + \alpha (\mathbf{u}^T)^T \mathbf{u}^T + \beta (\mathbf{v}^T)^T \mathbf{v}^T = \\ &= \mathbf{G}_k + \alpha \mathbf{u} \mathbf{u}^T + \beta \mathbf{v} \mathbf{v}^T.\end{aligned}$$

Следовательно, $\mathbf{G}_{k+1}^T = \mathbf{G}_{k+1}$. \square

Свойство 2. Если в формуле ДФП (4.20) матрица \mathbf{G}_k положительно определенная и $\mathbf{s}_k^T \mathbf{p}_k > 0$, то матрица \mathbf{G}_{k+1} также положительно определенная.

Доказательство. Пусть матрица \mathbf{G}_k положительно определенная и $\mathbf{x}^T \mathbf{G}_k \mathbf{x} > 0 \quad \forall \mathbf{x} \neq \mathbf{0}$. Тогда по формуле ДФП (4.20) имеем

$$\mathbf{x}^T \mathbf{G}_{k+1} \mathbf{x} = \mathbf{x}^T \mathbf{G}_k \mathbf{x} + \frac{(\mathbf{x}^T \mathbf{s}_k)^2}{\mathbf{s}_k^T \mathbf{p}_k} - \frac{(\mathbf{x}^T \mathbf{G}_k \mathbf{p}_k)^2}{\mathbf{p}_k^T \mathbf{G}_k \mathbf{p}_k}.$$

Обозначая $\mathbf{a} = \mathbf{G}_k^{1/2} \mathbf{x}$ и $\mathbf{b} = \mathbf{G}_k^{1/2} \mathbf{p}_k$, получим

$$\mathbf{x}^T \mathbf{G}_{k+1} \mathbf{x} = \frac{(\mathbf{a}^T \mathbf{a})(\mathbf{b}^T \mathbf{b}) - (\mathbf{a}^T \mathbf{b})^2}{\mathbf{b}^T \mathbf{b}} + \frac{(\mathbf{x}^T \mathbf{s}_k)^2}{\mathbf{s}_k^T \mathbf{p}_k}.$$

В правой части этого равенства первое слагаемое неотрицательно в силу неравенства Коши – Буняковского – Шварца $|\mathbf{a}^T \mathbf{b}| \leq \|\mathbf{a}\| \cdot \|\mathbf{b}\|$, а

второе слагаемое неотрицательно, если $\mathbf{s}_k^T \mathbf{p}_k > 0$. Покажем, что эти слагаемые не могут одновременно обращаться в нуль. Если первое слагаемое равно нулю, то $\mathbf{a} = \alpha \mathbf{b}$, а, значит, и $\mathbf{x} = \alpha \mathbf{p}_k$ при $\alpha \neq 0$. Но тогда по условию $\mathbf{x}^T \mathbf{s}_k = \alpha \mathbf{p}_k^T \mathbf{s}_k \neq 0$. Поэтому $\mathbf{x}^T \mathbf{G}_{k+1} \mathbf{x} > 0 \quad \forall \mathbf{x} \neq \mathbf{0}$. \square

Заметим, что условие этого свойства $\mathbf{s}_k^T \mathbf{p}_k > 0$ выполняется при условии точного одномерного поиска (3.7) в виде $\mathbf{s}_k^T \mathbf{g}_{k+1} = 0$ и обеспечения \mathbf{s}_k как направления спуска, для которого $\mathbf{s}_k^T \mathbf{g}_k < 0$. Действительно, в силу равенств (4.22) и (4.24) $\mathbf{s}_k = -\lambda_k \mathbf{G}_k \mathbf{g}_k$, $\lambda_k > 0$. Поэтому

$$\mathbf{s}_k^T \mathbf{p}_k = \mathbf{s}_k^T (\mathbf{g}_{k+1} - \mathbf{g}_k) = \mathbf{s}_k^T \mathbf{g}_{k+1} - \mathbf{s}_k^T \mathbf{g}_k = -\mathbf{s}_k^T \mathbf{g}_k = \lambda_k \mathbf{g}_k^T \mathbf{G}_k \mathbf{g}_k > 0.$$

Свойство сохранения положительной определенности матрицы \mathbf{G}_k гарантирует, что направление \mathbf{d}_k является направлением спуска.

Свойство 3. При минимизации квадратичной функции с положительно определенной матрицей Гессе \mathbf{H} методом ДФП с точным одномерным поиском выполняются равенства:

$$\mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i = \mathbf{s}_i, \quad i = \overline{0, k}; \quad (4.25)$$

$$\mathbf{s}_i^T \mathbf{H} \mathbf{s}_k = 0, \quad i = \overline{0, k-1}. \quad (4.26)$$

Доказательство. Применим метод математической индукции. Из квазиньютоновского условия (4.5) в виде $\mathbf{G}_1 \mathbf{p}_0 = \mathbf{s}_0$ с учетом свойства квадратичной функции (4.4) в виде $\mathbf{p}_0 = \mathbf{H} \mathbf{s}_0$ имеем $\mathbf{G}_1 \mathbf{H} \mathbf{s}_0 = \mathbf{s}_0$. Отсюда по формулам (4.24) с точным одномерным поиском получим:

$$\mathbf{s}_0^T \mathbf{H} \mathbf{s}_1 = \mathbf{s}_0^T \mathbf{H} \lambda_1 \mathbf{d}_1 = \mathbf{s}_0^T \mathbf{H} (-\lambda_1 \mathbf{G}_1 \mathbf{g}_1) = -\lambda_1 (\mathbf{G}_1 \mathbf{H} \mathbf{s}_0)^T \mathbf{g}_1 = -\lambda_1 \mathbf{s}_0^T \mathbf{g}_1 = 0.$$

Таким образом, равенства (4.25) и (4.26) выполняются при начальных значениях $k=0$ и $k=1$ соответственно.

Предположим, что равенства (4.25) и (4.26) выполнены для некоторого k . Докажем, что они выполняются и для $k+1$. Для $i = \overline{0, k}$ с использованием равенств (4.24) получим:

$$\begin{aligned} \mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} &= \mathbf{s}_i^T \mathbf{H} \lambda_{k+1} \mathbf{d}_{k+1} = \lambda_{k+1} \mathbf{s}_i^T \mathbf{H} (-\mathbf{G}_{k+1} \mathbf{g}_{k+1}) = \\ &= -\lambda_{k+1} (\mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i)^T \mathbf{g}_{k+1} = -\lambda_{k+1} \mathbf{s}_i^T \mathbf{g}_{k+1}. \end{aligned}$$

Применяя формулу (4.18) для градиента квадратичной функции, имеем

$$\mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} = -\lambda_{k+1} \mathbf{s}_i^T \left(\mathbf{g}_{i+1} + \sum_{j=i+1}^k \mathbf{H} \mathbf{s}_j \right) = -\lambda_{k+1} \left(\mathbf{s}_i^T \mathbf{g}_{i+1} + \sum_{j=i+1}^k \mathbf{s}_i^T \mathbf{H} \mathbf{s}_j \right).$$

По условию точного одномерного поиска (3.7) в виде $\mathbf{s}_i^T \mathbf{g}_{i+1} = 0$ и сделанного предположения индукции (4.26) получим для $i = \overline{0, k}$ условие сопряженности $\mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} = 0$. Это и доказывает справедливость равенств (4.26) для произвольного k .

С учетом свойства квадратичной функции (4.4) в виде $\mathbf{p}_{k+1} = \mathbf{H} \mathbf{s}_{k+1}$, предположения индукции (4.25) и доказанного равенства $\mathbf{s}_i^T \mathbf{H} \mathbf{s}_{k+1} = 0$ для $i = \overline{0, k}$ имеем:

$$\mathbf{p}_{k+1}^T \mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i = \mathbf{p}_{k+1}^T \mathbf{s}_i = \mathbf{s}_{k+1}^T \mathbf{H} \mathbf{s}_i = 0.$$

Отсюда по формуле ДФП (4.20) получим:

$$\mathbf{G}_{k+2} \mathbf{H} \mathbf{s}_i = \mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i + \frac{\mathbf{s}_{k+1} (\mathbf{s}_{k+1}^T \mathbf{H} \mathbf{s}_i)}{\mathbf{s}_{k+1}^T \mathbf{p}_{k+1}} - \frac{\mathbf{G}_{k+1} \mathbf{p}_{k+1} (\mathbf{p}_{k+1}^T \mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i)}{\mathbf{p}_{k+1}^T \mathbf{G}_{k+1} \mathbf{p}_{k+1}} = \mathbf{G}_{k+1} \mathbf{H} \mathbf{s}_i,$$

то есть по предположениям индукции $\mathbf{G}_{k+2} \mathbf{H} \mathbf{s}_i = \mathbf{s}_i$ для $i = \overline{0, k}$. С учетом квазиньютоновского условия (4.5) в виде $\mathbf{G}_{k+2} \mathbf{p}_{k+1} = \mathbf{s}_{k+1}$ и свойства квадратичной функции (4.4) в виде $\mathbf{p}_{k+1} = \mathbf{H} \mathbf{s}_{k+1}$ имеем $\mathbf{G}_{k+2} \mathbf{H} \mathbf{s}_{k+1} = \mathbf{s}_{k+1}$. Итак $\mathbf{G}_{k+2} \mathbf{H} \mathbf{s}_i = \mathbf{s}_i$ для $i = \overline{0, k+1}$, что доказывает

справедливость равенств (4.25) для произвольного k . \square

Это свойство показывает, что в силу равенств (4.26) метод ДФП является методом сопряженных направлений, поэтому он минимизирует квадратичную функцию с положительно определенной матрицей Гессе \mathbf{H} при точном одномерном поиске не более чем за n итераций.

Свойство 4. При минимизации квадратичной функции с положительно определенной матрицей Гессе \mathbf{H} методом ДФП с точным одномерным поиском после n итераций $\mathbf{G}_n = \mathbf{H}^{-1}$.

Доказательство. При сделанных предположениях после n итераций метода ДФП в силу выполнения равенств (4.26) при $k = n-1$ векторы $\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{n-1}$ являются сопряженными. Поэтому по лемме 3.1 они линейно независимы. Представим их столбцами невырожденной матрицы \mathbf{S} . Поскольку при этом выполняются равенства (4.25) при $k = n-1$ в виде $\mathbf{G}_n \mathbf{H} \mathbf{s}_i = \mathbf{s}_i$ для $i = \overline{0, n-1}$, то имеем $\mathbf{G}_n \mathbf{H} \mathbf{S} = \mathbf{S}$. Умножая это равенство справа на \mathbf{S}^{-1} , придем к равенству $\mathbf{G}_n \mathbf{H} = \mathbf{E}$, откуда получим $\mathbf{G}_n = \mathbf{H}^{-1}$. Это означает, что после n итераций метода ДФП аппроксимация обратной матрицы Гессе совпадает с ней. \square

Свойство 5. При минимизации квадратичной функции с положительно определенной матрицей Гессе \mathbf{H} методом ДФП с точным одномерным поиском после n итераций

$$\mathbf{H}^{-1} = \sum_{i=0}^{n-1} \frac{\mathbf{s}_i \mathbf{s}_i^T}{\mathbf{s}_i^T \mathbf{p}_i}. \quad (4.27)$$

Доказательство. После n итераций метода ДФП в силу равенств (4.26) при $k = n-1$ векторы $\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{n-1}$ являются сопряженными и линейно независимыми. Сформируем из них невырожденную матрицу \mathbf{S} . Из условий сопряженности (4.26) $\mathbf{S}^T \mathbf{H} \mathbf{S} = \mathbf{D}$, где \mathbf{D} – диагональная матрица с элементами $\mathbf{s}_i^T \mathbf{H} \mathbf{s}_i$. Поэтому $\mathbf{H} = (\mathbf{S}^T)^{-1} \mathbf{D} \mathbf{S}^{-1}$. Тогда $\mathbf{H}^{-1} = \mathbf{S} \mathbf{D}^{-1} \mathbf{S}^T$, где \mathbf{D}^{-1} – диагональная матрица с элементами

$1/(\mathbf{s}_i^T \mathbf{H} \mathbf{s}_i)$. Перемножим матрицы в правой части последнего равенства

$$\mathbf{H}^{-1} = \sum_{i=0}^{n-1} \frac{\mathbf{s}_i \mathbf{s}_i^T}{\mathbf{s}_i^T \mathbf{H} \mathbf{s}_i}.$$

Отсюда с использованием свойства квадратичной функции (4.4) в виде $\mathbf{p}_i = \mathbf{H} \mathbf{s}_i$ получим равенство (4.27). \square

Представим формулу ДФП (4.20) в виде

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \mathbf{A}_k - \mathbf{B}_k,$$

где

$$\mathbf{A}_k = \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{p}_k}, \quad \mathbf{B}_k = \frac{\mathbf{G}_k \mathbf{p}_k (\mathbf{G}_k \mathbf{p}_k)^T}{(\mathbf{G}_k \mathbf{p}_k)^T \mathbf{p}_k}.$$

Тогда

$$\mathbf{G}_n = \mathbf{G}_0 + \sum_{k=0}^{n-1} \mathbf{A}_k - \sum_{k=0}^{n-1} \mathbf{B}_k.$$

Для квадратичной функции по свойствам 4 и 5 имеем:

$$\mathbf{H}^{-1} = \mathbf{G}_0 + \sum_{k=0}^{n-1} \mathbf{A}_k - \sum_{k=0}^{n-1} \mathbf{B}_k, \quad \mathbf{H}^{-1} = \sum_{k=0}^{n-1} \mathbf{A}_k, \quad \mathbf{G}_0 = \sum_{k=0}^{n-1} \mathbf{B}_k.$$

Отсюда следует, что начальное задание аппроксимирующей матрицы $\mathbf{G}_0 = \mathbf{E}$ в процессе минимизации квадратичной функции компенсируются последней дробью в формуле (4.20).

Приведенные свойства метода Девидона – Флетчера – Пауэлла проявляются и при минимизации дифференцируемой целевой функции общего вида.

Вычислительные эксперименты, проведенные многими исследователями, показали, что метод Девидона – Флетчера – Пауэлла очень чувствителен к точности одномерного поиска. Если одномерная минимизация целевой функции проводится с невысокой точностью, то эффективность этого метода снижается.

4.6. Метод Бroyдена – Флетчера – Гольдфарба – Шанно

Пусть \mathbf{H}_k – аппроксимация матрицы Гессе $\mathbf{H}(\mathbf{x}) = \nabla^2 f(\mathbf{x})$ в точке \mathbf{x}_k , причем она связана с аппроксимацией обратной матрицы Гессе равенством $\mathbf{H}_k = \mathbf{G}_k^{-1}$. Умножая квазиньютоновское условие (4.5) слева на матрицу \mathbf{H}_{k+1} , после преобразований получим

$$\mathbf{H}_{k+1} \mathbf{s}_k = \mathbf{p}_k. \quad (4.28)$$

Эта формула отличается от формулы (4.5) тем, что матрица \mathbf{G}_{k+1} заменена матрицей \mathbf{H}_{k+1} , а векторы \mathbf{p}_k и \mathbf{s}_k поменялись местами. Выполним те же изменения в формуле ДФП (4.20)

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{p}_k \mathbf{p}_k^T}{\mathbf{p}_k^T \mathbf{s}_k} - \frac{\mathbf{H}_k \mathbf{s}_k (\mathbf{H}_k \mathbf{s}_k)^T}{(\mathbf{H}_k \mathbf{s}_k)^T \mathbf{s}_k}. \quad (4.29)$$

Эта формула называется *формулой Бroyдена – Флетчера – Гольдфарба – Шанно* (БФГШ), а использующий ее квазиньютоновский метод называется *методом Бroyдена – Флетчера – Гольдфарба – Шанно*. Метод БФГШ, основанный на формуле (4.29), представлен в 1970 году независимо английскими математиками Ч. Д. Бройденом и Р. Флетчером, американскими математиками Д. Гольдфарбом и Д. Ф. Шанно. Поскольку здесь аппроксимируется матрица Гессе, то направление одномерного поиска \mathbf{d}_k необходимо вычислять не по формуле (4.7), а путем решения системы линейных алгебраических уравнений

$$\mathbf{H}_k \mathbf{d}_k = -\mathbf{g}_k. \quad (4.30)$$

Первая итерация начинается из заданной начальной точки \mathbf{x}_0 , и в направлении антиградиента выполняется одномерный поиск:

$$\mathbf{H}_0 = \mathbf{E}, \quad \mathbf{d}_0 = -\mathbf{g}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0), \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0. \quad (4.31)$$

Последующие итерации проводятся с учетом формул (4.29) и (4.30):

$$\mathbf{p}_{k-1} = \mathbf{g}_k - \mathbf{g}_{k-1}, \quad \mathbf{s}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}, \quad (4.32)$$

$$\mathbf{v}_{k-1} = \mathbf{H}_{k-1} \mathbf{s}_{k-1}, \quad \mathbf{H}_k = \mathbf{H}_{k-1} + \frac{\mathbf{p}_{k-1} \mathbf{p}_{k-1}^T}{\mathbf{p}_{k-1}^T \mathbf{s}_{k-1}} - \frac{\mathbf{v}_{k-1} \mathbf{v}_{k-1}^T}{\mathbf{v}_{k-1}^T \mathbf{s}_{k-1}}, \quad (4.33)$$

$$\mathbf{H}_k \mathbf{d}_k = -\mathbf{g}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k. \quad (4.34)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon.$$

По формулам (4.31)–(4.34) составим алгоритм метода Бройдена – Флетчера – Гольдфарба – Шанно.

Алгоритм метода БФГШ.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – процедура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$ и положить $\mathbf{d} = -\mathbf{g}_x$, $\mathbf{H} = \mathbf{E}$.

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\mathbf{g}_y = \mathbf{g}_x$.

4. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, $\mathbf{p} = \mathbf{g}_x - \mathbf{g}_y$, $\mathbf{v} = \mathbf{H} \cdot \mathbf{s}$.

5. Положить $\mathbf{H} = \mathbf{H} + \mathbf{p} \cdot \mathbf{p}^T / (\mathbf{p}^T \cdot \mathbf{s}) - \mathbf{v} \cdot \mathbf{v}^T / (\mathbf{v}^T \cdot \mathbf{s})$.

6. Решить СЛАУ $\mathbf{H} \cdot \mathbf{d} = -\mathbf{g}_x$.

7. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 2.

8. Остановиться.

Пример 4.5. Для вычисления точки минимума квадратичной функции (1.3) с допустимой погрешностью 10^{-3} метод БФГШ затратил 3 итерации и 19 вычислений функции. Траектория поиска такая же, как и траектория метода Бройдена из примера 4.1, представленная на рис. 4.1.

Пример 4.6. На рис. 4.4 показана траектория минимизации функции Розенброка методом БФГШ. Вычисление точки минимума с допустимой погрешностью 10^{-3} потребовало 17 итераций и 265 вычислений функции, что полностью совпало с результатами метода ДФП из примера 4.4. Траектории поиска на рис. 4.4 и рис. 4.3 также совпадают.

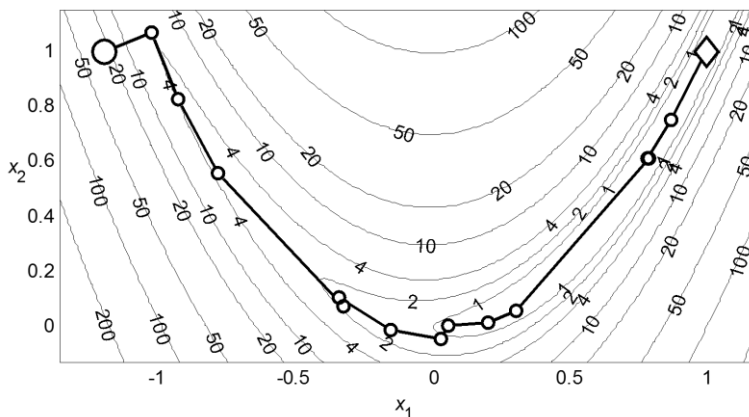


Рис. 4.4. Минимизация функции Розенброка методом БФГШ

Этот метод обладает теми же свойствами, что и метод ДФП, но он менее чувствителен к точности одномерного поиска.

4.7. Модификация метода БФГШ

Недостатком метода Бройдена – Флетчера – Гольдфарба – Шанно по сравнению с методом Девидона – Флетчера – Пауэлла является необходимость решения системы линейных алгебраических уравнений (4.30) для определения вектора направления одномерного поиска \mathbf{d}_k . Если коррекция некоторой квадратной матрицы \mathbf{A} проводится поправкой ранга один, то обратную матрицу можно найти по формуле Шермана – Моррисона

$$(\mathbf{A} + \mathbf{a}\mathbf{b}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{a}\mathbf{b}^T\mathbf{A}^{-1}}{1 + \mathbf{b}^T\mathbf{A}^{-1}\mathbf{a}}.$$

Дважды применяя эту формулу к формуле БФГШ (4.29), получим модификацию этой формулы

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \left(1 + \frac{\mathbf{p}_k^T \mathbf{G}_k \mathbf{p}_k}{\mathbf{s}_k^T \mathbf{p}_k}\right) \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{p}_k} - \frac{\mathbf{s}_k (\mathbf{G}_k \mathbf{p}_k)^T + \mathbf{G}_k \mathbf{p}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{p}_k}. \quad (4.35)$$

Эта формула называется *модифицированной формулой Бройдена – Флетчера – Гольдфарба – Шанно*, а использующий ее квазиньютоновский метод называется *модифицированным методом БФГШ*. Вывод формулы (4.35) приведен в подразделе 5.4.

На первой итерации из заданной начальной точки \mathbf{x}_0 выполняется одномерный поиск в направлении антиградиента:

$$\mathbf{G}_0 = \mathbf{E}, \quad \mathbf{d}_0 = -\mathbf{g}_0, \quad \lambda_0 = \arg \min_{\lambda} f(\mathbf{x}_0 + \lambda \mathbf{d}_0), \quad \mathbf{x}_1 = \mathbf{x}_0 + \lambda_0 \mathbf{d}_0. \quad (4.36)$$

Последующие итерации выполняются с учетом формулы (4.35):

$$\mathbf{p}_{k-1} = \mathbf{g}_k - \mathbf{g}_{k-1}, \quad \mathbf{s}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}, \quad \mathbf{v}_{k-1} = \mathbf{G}_{k-1} \mathbf{p}_{k-1}, \quad (4.37)$$

$$\mathbf{G}_k = \mathbf{G}_{k-1} + \left(1 + \frac{\mathbf{p}_{k-1}^T \mathbf{v}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{p}_{k-1}}\right) \frac{\mathbf{s}_{k-1} \mathbf{s}_{k-1}^T}{\mathbf{s}_{k-1}^T \mathbf{p}_{k-1}} - \frac{\mathbf{s}_{k-1} \mathbf{v}_{k-1}^T + \mathbf{v}_{k-1} \mathbf{s}_{k-1}^T}{\mathbf{s}_{k-1}^T \mathbf{p}_{k-1}}, \quad (4.38)$$

$$\mathbf{d}_k = -\mathbf{G}_k \mathbf{g}_k, \quad \lambda_k = \arg \min_{\lambda} f(\mathbf{x}_k + \lambda \mathbf{d}_k), \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k. \quad (4.39)$$

Итерации продолжаются до тех пор, пока выполняется условие

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\| > \varepsilon.$$

По формулам (4.36)–(4.39) составим алгоритм модифицированного метода Бройдена – Флетчера – Гольдфарба – Шанно.

Алгоритм модифицированного метода БФГШ.

Входные параметры: \mathbf{x} – начальная точка поиска, $f(\mathbf{x})$ – проце-

дура вычисления функции, ε – допустимая погрешность.

Выходной параметр \mathbf{x} – конечная точка поиска.

1. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$ и положить $\mathbf{d} = -\mathbf{g}_x$, $\mathbf{G} = \mathbf{E}$.

2. Вычислить $r = \arg \min_{\lambda} f(\mathbf{x} + \lambda \cdot \mathbf{d})$, $\mathbf{s} = r \cdot \mathbf{d}$.

3. Положить $\mathbf{x} = \mathbf{x} + \mathbf{s}$, $\mathbf{g}_y = \mathbf{g}_x$.

4. Вычислить $\mathbf{g}_x = \nabla f(\mathbf{x})$, $\mathbf{p} = \mathbf{g}_x - \mathbf{g}_y$, $\mathbf{v} = \mathbf{G} \cdot \mathbf{p}$, $\alpha = 1/\mathbf{s}^T \cdot \mathbf{p}$.

5. Положить $\mathbf{G} = \mathbf{G} + (1 + \alpha \cdot \mathbf{p}^T \cdot \mathbf{v}) \cdot \alpha \cdot \mathbf{s} \cdot \mathbf{s}^T - \alpha \cdot (\mathbf{s} \cdot \mathbf{v}^T + \mathbf{v} \cdot \mathbf{s}^T)$.

6. Вычислить $\mathbf{d} = -\mathbf{G} \cdot \mathbf{g}_x$.

7. Если $\|\mathbf{s}\| > \varepsilon$, то перейти к шагу 2.

8. Остановиться.

Этот алгоритм не требует решения системы линейных алгебраических уравнений в отличие от предыдущего алгоритма.

Пример 4.7. Для вычисления точки минимума квадратичной функции (1.3) с допустимой погрешностью $\varepsilon = 10^{-3}$ модифицированный метод БФГШ затратил 3 итерации и 19 вычислений функции. Траектория поиска такая же, как и траектория метода Бройдена из примера 4.1, представленная на рис. 4.1.

Пример 4.8. Минимизация функции Розенброка модифицированным методом БФГШ с погрешностью $\varepsilon = 10^{-3}$ и допустимой погрешностью одномерного поиска $\varepsilon_1 = 10^{-5}$ потребовала 17 итераций и 265 вычислений функции, как и в примере 4.6, но время вычислений уменьшилось благодаря исключению решения СЛАУ. Траектория поиска такая же, как и на рис. 4.4. На рис. 4.5 представлена траектория минимизации функции Розенброка модифицированным методом БФГШ при $\varepsilon = 10^{-3}$ и $\varepsilon_1 = 10^{-2}$, что потребовало 26 итераций и 166 вычислений функции. При снижении точности одномерного поиска число итераций возросло, но количество вычислений функции уменьшилось. Этот пример подтверждает слабую чувствительность метода БФГШ к точности одномерного поиска.

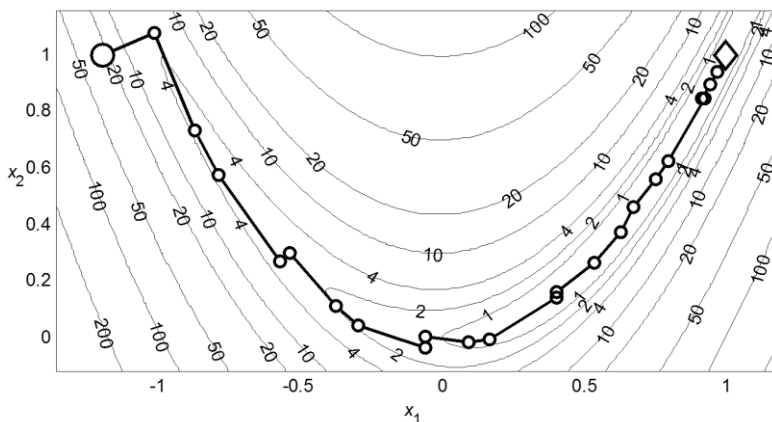


Рис. 4.5. Минимизация функции Розенброка модифицированным методом БФГШ

4.8. Сравнение методов безусловной оптимизации

Методы многомерной безусловной оптимизации, предназначенные для вычисления точки минимума \mathbf{x}^* унимодальной целевой функции многих переменных $f(\mathbf{x})$, по использованию значений производных функции делятся на три группы.

1. Методы нулевого порядка. Это метод циклического покоординатного спуска и метод сопряженных направлений Пауэлла, основанные на вычислении значений самой целевой функции и не использующие значений ее производных.

2. Методы первого порядка. Это метод наискорейшего спуска, методы сопряженных градиентов Флетчера – Ривса и Полака – Рибьера, квазиньютоновские методы Бroyдена, ДФП и БФГШ, использующие значения первых производных целевой функции.

3. Методы второго порядка. Это метод Ньютона, метод Ньютона с одномерным поиском, метод Ньютона с направлением спуска и метод Марквардта, использующие значения первых и вторых производных целевой функции.

Эти методы многомерного поиска можно сравнить по скорости сходимости. Методы многомерного поиска должны порождать последовательность точек $\{\mathbf{x}_k\}$, для которой

$$\lim_{k \rightarrow \infty} \mathbf{x}_k = \mathbf{x}^*.$$

Тогда по свойству предела

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| < \|\mathbf{x}_k - \mathbf{x}^*\|,$$

поэтому выполняется неравенство

$$\frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} < 1,$$

В этом случае говорят о *глобальной сходимости* метода. Например, если непрерывно дифференцируемая функция $f(\mathbf{x})$ удовлетворяет условию $f(\mathbf{x}) \rightarrow +\infty$ при $\|\mathbf{x}\| \rightarrow +\infty$, то для любой начальной точки \mathbf{x}_0 метод наискорейшего спуска сходится к стационарной точке функции $f(\mathbf{x})$. Глобальная сходимость методов сопряженных градиентов Флетчера – Ривса и Полака – Рибьера обеспечена лишь в случае процесса с периодической сменой начала, то есть с рестартами. Тогда их глобальная сходимость следует из глобальной сходимости метода наискорейшего спуска. Метод Ньютона не обладает свойством глобальной сходимости: если начальная точка \mathbf{x}_0 слишком далека от \mathbf{x}^* , то метод не сходится. Но методы Ньютона с одномерным поиском и с направлением спуска, а также метод Марквардта обладают глобальной сходимостью. Метод Бroyдена не имеет глобальной сходимости. Квазиньютоновские методы Девидона – Флетчера – Пауэлла и Бройдена – Флетчера – Гольдфарба – Шанно обладают глобальной сходимостью только в случае применений рестартов.

Если глобальная сходимость метода установлена, то вызывает интерес оценка ее эффективности. С практической точки зрения эффек-

тивность алгоритма зависит от числа итераций, необходимых для получения приближения оптимальной точки \mathbf{x}^* с допустимой погрешностью ε . Если допустить, что время вычисления итераций одинаково для всех алгоритмов, то наилучшим среди них будет тот, который требует наименьшего числа итераций.

Поведение последовательности точек $\{\mathbf{x}_k\}$ в окрестности оптимальной точки \mathbf{x}^* позволяет установить характер *асимптотической сходимости*. Если выполняется неравенство

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} = \alpha < 1$$

и $\alpha \neq 0$, то говорят, что имеет место *линейная сходимость* и что α – соответствующий *коэффициент сходимости*.

При дальнейшем сравнении скорости сходимости методов минимизации будем полагать, что целевая функция $f(\mathbf{x})$ дважды непрерывно дифференцируема и матрица Гессе $\nabla^2 f(\mathbf{x}^*)$ положительно определена. При этих условиях для метода наискорейшего спуска доказано, что последовательность $\{f(\mathbf{x}_k)\}$ удовлетворяет неравенству

$$\limsup_{k \rightarrow \infty} \frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} = \alpha \leq \frac{A - a}{A + a},$$

где A и a – соответственно наибольшее и наименьшее собственные значения матрицы Гессе $\nabla^2 f(\mathbf{x}^*)$ в точке \mathbf{x}^* . В худшем случае имеет место линейная сходимость, для которой коэффициент асимптотической сходимости $(A - a)/(A + a)$ непосредственно связан с числом обусловленности матрицы $\text{cond} \nabla^2 f(\mathbf{x}^*) = A/a$. Этот результат показывает, что сходимость метода наискорейшего спуска может быть очень медленной для плохо обусловленных функций овражного типа.

Если выполняется равенство

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} = 0,$$

то говорят, что имеет место *сверхлинейная сходимость*. При этом, если существует такое $\gamma > 1$, что

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^\gamma} < +\infty,$$

то говорят, что имеет место *сходимость порядка γ* . В частности, если

$$\lim_{k \rightarrow \infty} \lim_{k \rightarrow \infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^2} < +\infty,$$

то говорят, что имеет место *квадратичная сходимость*.

Методы сопряженных градиентов Флетчера – Ривса и Полака – Рибьера имеют сверхлинейную скорость сходимости по n шагам. Если матрица Гессе $\nabla^2 f(\mathbf{x})$ удовлетворяет условию Липшица, то для этих методов доказана квадратичная скорость сходимости по n шагам.

Для квазиньютоновских методов ДФП и БФГШ доказана сверхлинейная скорость сходимости. Если матрица Гессе $\nabla^2 f(\mathbf{x})$ удовлетворяет условию Липшица, то для этих методов доказана квадратичная скорость сходимости. Это показывает преимущество квазиньютоновских методов перед методами сопряженных градиентов, которые требуют приблизительно в n раз больше итераций для одного и того же асимптотического поведения. Однако это преимущество сильно снижается загрузкой памяти пропорционально n^2 и объемом промежуточных матричных вычислений пропорционально n^2 .

Метод Ньютона имеет квадратичную локальную скорость сходимости, если $\nabla^2 f(\mathbf{x})$ удовлетворяет в окрестности точки \mathbf{x}^* условию Липшица. Следовательно, в малой окрестности оптимальной точки метод Ньютона при указанных предположениях относительно целевой

функции $f(\mathbf{x})$ сходится быстрее остальных методов.

Однако при решении конкретных задач оптимизации необходимы вычислительные эксперименты для выбора самых эффективных методов, позволяющих надежно решать эти задачи с допустимой погрешностью при использовании минимальных вычислительных ресурсов.

Пример 4.9. В табл. 4.1–4.4 приведены число итераций и число вычислений функции при минимизации функции Розенброка с допустимой погрешностью $\varepsilon = 10^{-3}$ разными методами многомерной оптимизации: Ньютона с одномерным поиском (НОП), Ньютона с направлением спуска (ННС), Марквардта с одномерным поиском (МОП), Пауэлла (П), Полака – Рибьера (ПР), Бройдена (Б), ДФП и БФГШ. При этом для одномерного поиска использованы метод квадратичной интерполяции с тремя точками (И2) и метод кубической интерполяции с четырьмя точками (И3), а также различные значения допустимой погрешности одномерного поиска ε_1 . Очевидно, что эффективность многомерного метода существенно зависит от применяемого метода одномерного поиска и значения его допустимой погрешности. Для минимизации функции Розенброка самыми надежными и эффективными оказались методы Ньютона с одномерным поиском и БФГШ.

Таблица 4.1 – Число итераций с методом И2

ε_1	НОП	ННС	МОП	П	ПР	Б	ДФП	БФГШ
10^{-5}	12	12	15	12	17	17	17	17
10^{-4}	12	13	15	12	15	17	19	17
10^{-3}	12	15	16	12	–	20	–	20

Таблица 4.2 – Число вычислений функции с методом И2

ε_1	НОП	ННС	МОП	П	ПР	Б	ДФП	БФГШ
10^{-5}	260	307	398	529	266	269	265	265
10^{-4}	216	252	288	423	206	221	237	219
10^{-3}	163	200	228	304	–	193	–	199

Таблица 4.3 – Число итераций с методом ИЗ

ε_1	НОП	ННС	МОП	П	ПР	Б	ДФП	БФГШ
10^{-5}	12	12	15	12	17	17	17	17
10^{-4}	12	12	15	12	17	17	17	17
10^{-3}	12	11	15	–	20	20	–	20

Таблица 4.4 – Число вычислений функции с методом ИЗ

ε_1	НОП	ННС	МОП	П	ПР	Б	ДФП	БФГШ
10^{-5}	148	148	186	242	150	151	148	148
10^{-4}	140	138	178	230	138	140	137	136
10^{-3}	131	119	164	–	139	147	–	141

Лабораторная работа

Тема лабораторной работы.

Разработка подпрограмм для квазиньютоновских методов оптимизации.

Цель лабораторной работы.

Разработать подпрограмму квазиньютоновского метода безусловной минимизации функций многих переменных.

Порядок выполнения лабораторной работы.

1. Получить у преподавателя вариант целевой функции $f(\mathbf{x})$ многих переменных с начальной точкой поиска минимума \mathbf{x}_0 и задание на программирование одного из квазиньютоновских методов:

- 1) метод Бroyдена;
- 2) метод Девидона – Флетчера – Пауэлла;
- 3) метод Бroyдена – Флетчера – Гольдфарба – Шанно;
- 4) модифицированный метод Бroyдена – Флетчера – Гольдфарба – Шанно.

2. Получить у преподавателя задание на программирование одного из методов одномерного поиска в многомерном пространстве:

- 1) метод дихотомии;
- 2) метод деления интервала пополам;
- 3) метод Фибоначчи;
- 4) метод золотого сечения;
- 5) метод адаптации шага;
- 6) метод квадратичной интерполяции с тремя точками;
- 7) метод квадратичной интерполяции с двумя точками;
- 8) метод секущих;
- 9) метод касательных;
- 10) метод кубической интерполяции с четырьмя точками;
- 11) метод кубической интерполяции с двумя точками.

3. Аналитически найти точку \mathbf{x}^* минимума заданной функции $f(\mathbf{x})$ и вычислить минимальное значение функции $f^* = f(\mathbf{x}^*)$.

4. Написать подпрограмму вычисления значений функции $f(\mathbf{x})$ с входным параметром \mathbf{x} и выходным параметром f .

5. Протестировать подпрограмму вычисления функции $f(\mathbf{x})$ при вычислении значений $f_0 = f(\mathbf{x}_0)$ и $f^* = f(\mathbf{x}^*)$, сравнить f^* с f_0 .

6. На основании алгоритмов для заданного квазиньютоновского метода и метода одномерного поиска написать компьютерные подпрограммы в соответствии с требованиями:

1) входные параметры подпрограмм: \mathbf{x}_0 – начальная точка поиска, $f(\mathbf{x})$ – целевая функция, ε – допустимая погрешность;

2) выходные параметры подпрограмм: \mathbf{X} – массив всех точек поиска \mathbf{x}_k ; \mathbf{F} – массив соответствующих значений функции $f_k = f(\mathbf{x}_k)$; \mathbf{x} и $f_{\mathbf{x}}$ – лучшая точка поиска и значение в ней функции;

3) на каждой итерации выводить на экран компьютера строку, содержащую номер точки поиска k , шаг перехода в новую точку $h_k = \|\mathbf{s}_k\| = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|$, значение функции f_k и значение вектора переменных \mathbf{x}_k ;

4) итерации продолжаются до тех пор, пока выполняется неравен-

ство $h_k > \varepsilon$;

5) после окончания процесса оптимизации на экране должна отображаться таблица, имеющая соответствующий заголовок и представляющая процесс минимизации функции;

6) под таблицей необходимо отобразить количество вычислений целевой функции, конечный шаг поиска, минимальное значение функции и соответствующую ему точку поиска.

7. Написать общую программу для выполнения вычислительного процесса минимизации целевой функции многих переменных и отображения на экране этого процесса в соответствии с требованиями:

1) за основу общей программы для минимизации функции взять общую программу из предыдущей лабораторной работы и модифицировать эту программу;

2) задать в программе значения \mathbf{x}_0 , h , ε ;

3) путем вызова разработанной подпрограммы заданного квазиньютоновского метода с использованием подпрограмм вычисления функции $f(\mathbf{x})$ и заданного метода одномерного поиска, а также значений \mathbf{x}_0 , h , ε реализовать итерационный процесс минимизации функции с построением таблицы процесса на экране и получением массивов значений вектора переменных \mathbf{X} и функции \mathbf{F} , лучшей точки поиска \mathbf{x} и значения в ней функции $f_{\mathbf{x}}$;

4) с помощью подпрограммы графического отображения итерационного процесса по массивам \mathbf{X} и \mathbf{F} отобразить графики процесса минимизации целевой функции многих переменных на экране компьютера.

8. Выполнить анализ полученных результатов:

1) на основании табличной и графической информации о работе метода одномерного поиска в многомерном пространстве определить конечные точки метода Свенна и показать найденный этим методом интервал неопределенности;

2) на нескольких начальных итерациях по табличным данным и графикам сопоставить начало работы квазиньютоновского метода с

его рабочими формулами;

3) оценить эффективность квазиньютоновского метода и сопоставить ее с эффективностью методов многомерной безусловной оптимизации из предыдущих лабораторных работ.

9. Ответить на вопросы преподавателя по теме лабораторной работы.

10. Оформить и сдать преподавателю отчет о проведенной лабораторной работе.

С о д е р ж а н и е о т ч е т а :

1) титульный лист установленного образца с указанием организации, названия учебной дисциплины, темы работы, номера варианта, исполнителя и принимающего, города, года;

2) постановка задачи минимизации целевой функции, включающая задание функции, начальную точку и точку минимума;

3) теоретическое описание квазиньютоновского метода безусловной минимизации функций многих переменных;

4) алгоритм квазиньютоновского метода;

5) теоретическое описание метода одномерного поиска в многомерном пространстве;

6) алгоритм метода одномерного поиска;

7) распечатки используемых компьютерных программ;

8) табличное представление процесса минимизации и его итоги – количество вычислений значений целевой функции, минимальное значение функции и соответствующее ему значение точки минимума, достигнутую точность;

9) трехмерные графики функции и траектории поиска со всеми точками;

10) двумерные графики линий уровня функции и траектории поиска со всеми точками;

11) анализ полученных результатов;

12) выводы о проделанной лабораторной работе.

Контрольные вопросы

1. Какие достоинства и недостатки метода Ньютона с одномерным поиском привели к открытию квазиньютоновских методов?
2. Дайте определение квазиньютоновских методов.
3. Выведите квазиньютоновское условие, на котором основаны квазиньютоновские методы.
4. Запишите уравнение коррекции для аппроксимации обратной матрицы Гессе.
5. Дайте определение поправки для аппроксимации обратной матрицы Гессе.
6. Как определяется направление одномерного поиска в квазиньютоновских методах?
7. Запишите итерационные формулы квазиньютоновских методов.
8. Опишите квазиньютоновские методы.
9. Как вычисляется величина шага в квазиньютоновских методах?
10. Как заканчиваются вычисления в квазиньютоновских методах?
11. Какого порядка квазиньютоновские методы?
12. Укажите достоинства и недостатки квазиньютоновских методов.
13. Какое другое название квазиньютоновских методов вы знаете?
14. Приведите формулы поправки ранга один.
15. Выведите формулу Бройдена.
16. Запишите итерационные формулы метода Бройдена.
17. Опишите метод Бройдена.
18. Как определяется направление одномерного поиска в методе Бройдена?
19. Как вычисляется величина шага в методе Бройдена?
20. Как заканчиваются вычисления в методе Бройдена?
21. Какого порядка метод Бройдена?
22. Составьте алгоритм метода Бройдена.
23. Укажите достоинства и недостатки метода Бройдена.

24. Приведите формулы поправки ранга два.
25. Выведите формулу Девидона – Флетчера – Пауэлла.
26. Запишите итерационные формулы метода ДФП.
27. Опишите метод Девидона – Флетчера – Пауэлла.
28. Как определяется направление одномерного поиска в методе Девидона – Флетчера – Пауэлла?
29. Как вычисляется величина шага в методе ДФП?
30. Как заканчиваются вычисления в методе ДФП?
31. Какого порядка метод Девидона – Флетчера – Пауэлла?
32. Составьте алгоритм метода Девидона – Флетчера – Пауэлла.
33. Укажите достоинства и недостатки метода ДФП.
34. На какой функции проявляются свойства метода Девидона – Флетчера – Пауэлла?
35. Приведите свойства метода Девидона – Флетчера – Пауэлла.
36. Выведите формулу БФГШ.
37. Запишите итерационные формулы метода БФГШ.
38. Опишите метод Бройдена – Флетчера – Гольдфарба – Шанно.
39. Как определяется направление одномерного поиска в методе Бройдена – Флетчера – Гольдфарба – Шанно?
40. Как вычисляется величина шага в методе БФГШ?
41. Как заканчиваются вычисления в методе БФГШ?
42. Какого порядка метод БФГШ?
43. Составьте алгоритм метода БФГШ.
44. Укажите достоинства и недостатки метода БФГШ.
45. На какой функции проявляются свойства метода БФГШ?
46. Приведите свойства метода БФГШ.
47. Приведите формулу Шермана – Моррисона.
48. Приведите модифицированную формулу БФГШ.
49. Запишите итерационные формулы модифицированного метода Бройдена – Флетчера – Гольдфарба – Шанно.
50. Опишите модифицированный метод БФГШ.
51. Как определяется направление одномерного поиска в модифицированном методе Бройдена – Флетчера – Гольдфарба – Шанно?

52. Как вычисляется величина шага в модифицированном методе Бройдена – Флетчера – Гольдфарба – Шанно?
53. Как заканчиваются вычисления в модифицированном методе Бройдена – Флетчера – Гольдфарба – Шанно?
54. Какого порядка модифицированный метод БФГШ?
55. Составьте алгоритм модифицированного метода Бройдена – Флетчера – Гольдфарба – Шанно.
56. Укажите достоинства и недостатки модифицированного метода Бройдена – Флетчера – Гольдфарба – Шанно.
57. На какие группы можно разделить методы многомерной безусловной оптимизации?
58. Дайте определение глобальной сходимости метода многомерной безусловной оптимизации.
59. Дайте определение асимптотической сходимости метода многомерной безусловной оптимизации.
60. Как определяется линейная сходимость метода многомерной безусловной оптимизации?
61. Приведите примеры методов многомерной безусловной оптимизации, обладающих свойством линейной сходимости.
62. Как определяется сверхлинейная сходимость метода многомерной безусловной оптимизации?
63. Приведите примеры методов многомерной безусловной оптимизации, обладающих свойством сверхлинейной сходимости?
64. Как определяется квадратичная сходимость метода многомерной безусловной оптимизации?
65. Приведите примеры методов многомерной безусловной оптимизации, обладающих свойством квадратичной сходимости?
66. Чем определяется выбор метода многомерной безусловной оптимизации для решения конкретной задачи оптимизации?

5. СПРАВОЧНЫЙ МАТЕРИАЛ

Освоение теории и практики методов многомерной безусловной минимизации требует знания многих понятий и формул математического анализа, высшей алгебры и теории матриц. В этом разделе обобщается основной справочный материал, необходимый для изучения методов многомерной оптимизации. Приводятся правила дифференцирования функций. Рассматриваются элементы векторной алгебры. Представляются действия с матрицами. Приводится формула Шермана – Моррисона для обращения матрицы, на основании которой выводится формула модифицированного метода Бroyдена – Флетчера – Гольдфарба – Шанно. Рассматривается дифференцирование функций многих переменных. Приводится формула Тейлора для функции многих переменных. Рассматриваются квадратичные формы, формулируется критерий Сильвестра-Якоби.

5.1. Правила дифференцирования

При аналитическом нахождении точки минимума функции необходимо использовать формулы дифференциального исчисления функций одной переменной. Ниже приведены основные правила дифференцирования функций, а также таблица производных основных элементарных функций.

Производная постоянной величины $C' = 0$.

Производная суммы $(u + v)' = u' + v'$.

Производная разности $(u - v)' = u' - v'$.

Постоянный множитель можно выносить за знак производной:

$$(Cy)' = Cy'.$$

Производная произведения $(uv)' = u'v + uv'$.

Производная частного

$$\left(\frac{u}{v}\right)' = \frac{u'v - uv'}{v^2}.$$

Производная сложной функции $[f(u(x))]' = f'_u \cdot u'_x$.

Производная обратной функции

$$x'_y = \frac{1}{y'_x}.$$

Производная функции, заданной параметрически:

$$\begin{cases} x = x(t), \\ y = y(t); \end{cases} \quad \frac{dy}{dx} = \frac{y'_t}{x'_t}.$$

Производные основных элементарных функций даны в табл. 5.1. В этой таблице a – некоторый числовой постоянный параметр. Структура таблицы позволяет установить аналогию и отличие между производными тригонометрических и гиперболических функций, а также производными обратных тригонометрических и обратных гиперболических функций. Здесь обозначены гиперболические и обратные гиперболические функции:

$$\operatorname{sh} x = \frac{e^x - e^{-x}}{2}, \quad \operatorname{ch} x = \frac{e^x + e^{-x}}{2},$$

$$\operatorname{th} x = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad \operatorname{cth} x = \frac{e^x + e^{-x}}{e^x - e^{-x}},$$

$$\operatorname{Arsh} x = \ln(x + \sqrt{x^2 + 1}), \quad \operatorname{Arch} x = \ln(x + \sqrt{x^2 - 1}),$$

$$\operatorname{Arth} x = \frac{1}{2} \ln \frac{1+x}{1-x}, \quad \operatorname{Arch} x = \frac{1}{2} \ln \frac{1+x}{1-x}.$$

Таблица 5.1 – Производные элементарных функций

$f(x)$	$f'(x)$	$f(x)$	$f'(x)$
x^a	ax^{a-1}	a^x	$a^x \ln a$
$\ln x$	$\frac{1}{x}$	e^x	e^x
$\sin x$	$\cos x$	$\arcsin x$	$\frac{1}{\sqrt{1-x^2}}$
$\cos x$	$-\sin x$	$\arccos x$	$-\frac{1}{\sqrt{1-x^2}}$
$\operatorname{tg} x$	$\frac{1}{\cos^2 x}$	$\operatorname{arctg} x$	$\frac{1}{1+x^2}$
$\operatorname{ctg} x$	$-\frac{1}{\sin^2 x}$	$\operatorname{arctg} x$	$-\frac{1}{1+x^2}$
$\operatorname{sh} x$	$\operatorname{ch} x$	$\operatorname{Arsh} x$	$\frac{1}{\sqrt{x^2+1}}$
$\operatorname{ch} x$	$\operatorname{sh} x$	$\operatorname{Arch} x$	$\frac{1}{\sqrt{x^2-1}}$
$\operatorname{th} x$	$\frac{1}{\operatorname{ch}^2 x}$	$\operatorname{Arth} x$	$\frac{1}{1-x^2}$
$\operatorname{cth} x$	$-\frac{1}{\operatorname{sh}^2 x}$	$\operatorname{Arth} x$	$\frac{1}{1-x^2}$

Логарифмическое дифференцирование:

$$1) \ y = u^v, \ \ln y = v \ln u, \ (\ln y)' = \frac{1}{y} y', \ y' = y(v \ln u)',$$

$$y' = y \left(v' \ln u + v \frac{1}{u} u' \right);$$

$$2) \quad y = \frac{u_1 \cdot u_2 \cdot \dots \cdot u_n}{v_1 \cdot v_2 \cdot \dots \cdot v_n},$$

$$\ln y = \ln u_1 + \ln u_2 + \dots + \ln u_n - \ln v_1 - \ln v_2 - \dots - \ln v_n,$$

$$y' = y \left(\frac{1}{u_1} u_1' + \frac{1}{u_2} u_2' + \dots + \frac{1}{u_n} u_n' - \frac{1}{v_1} v_1' - \frac{1}{v_2} v_2' - \dots - \frac{1}{v_n} v_n' \right).$$

5.2. Элементы векторной алгебры

Вектором называется упорядоченное множество действительных чисел. Например, $\mathbf{a} = (a_1, a_2, \dots, a_n)$, где a_1, a_2, \dots, a_n действительные числа, есть вектор, состоящий из n элементов. Количество элементов n называется *размером* вектора. Числа a_1, a_2, \dots, a_n называются *проекциями* вектора.

Два вектора $\mathbf{a} = (a_1, a_2, \dots, a_n)$ и $\mathbf{b} = (b_1, b_2, \dots, b_n)$ называются *равными*, если равны все их соответствующие проекции: $a_i = b_i$, $i = \overline{1, n}$. В этом случае пишут $\mathbf{a} = \mathbf{b}$.

Если $\mathbf{a} = (a_1, a_2, \dots, a_n)$ и $\mathbf{b} = (b_1, b_2, \dots, b_n)$ то определены операции сложения и вычитания векторов:

$$\mathbf{a} + \mathbf{b} = (a_1 + b_1, a_2 + b_2, \dots, a_n + b_n),$$

$$\mathbf{a} - \mathbf{b} = (a_1 - b_1, a_2 - b_2, \dots, a_n - b_n).$$

Для любой скалярной величины α определена операция умножения вектора на скаляр $\alpha \mathbf{a} = (\alpha a_1, \alpha a_2, \dots, \alpha a_n)$.

Вектор $\mathbf{0} = (0, 0, \dots, 0)$ называется нулевым вектором.

Векторы $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ называют *линейно зависимыми*, если существуют скаляры $\alpha_1, \alpha_2, \dots, \alpha_m$, не все равные нулю, что

$$\alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2 + \dots + \alpha_m \mathbf{a}_m = \mathbf{0}.$$

В этом случае хотя бы один из этих векторов можно представить *линейной комбинацией* остальных векторов, например, при $\alpha_m \neq 0$

$$\mathbf{a}_m = \lambda_1 \mathbf{a}_1 + \lambda_2 \mathbf{a}_2 + \dots + \lambda_{m-1} \mathbf{a}_{m-1}.$$

В противном случае векторы называют *линейно независимыми*.

Множество всех n -мерных векторов называется n -мерным *векторным пространством* и обозначается R^n . Говорят, что векторное пространство R^n *натянута* на некоторую систему векторов, если каждый вектор из R^n можно представить в виде линейной комбинации этой системы. *Базис* векторного пространства R^n определяется как система n линейно независимых векторов. Элемент $\mathbf{x} = (x_1, x_2, \dots, x_n)$ пространства R^n также называется *точкой*. В этом случае x_1, x_2, \dots, x_n называются *координатами* точки.

Скалярное произведение двух векторов есть число, равное

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + \dots + a_n b_n.$$

Евклидова норма вектора \mathbf{a} определяется равенством

$$\|\mathbf{a}\| = \sqrt{\mathbf{a} \cdot \mathbf{a}} = \sqrt{a_1^2 + a_2^2 + \dots + a_n^2}$$

и соответствует длине вектора. Ее также обозначают $|\mathbf{a}|$ и называют *модулем* вектора.

Если векторы \mathbf{a} и \mathbf{b} рассматриваются как направленные отрезки, между которыми угол θ , то

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| \cdot |\mathbf{b}| \cdot \cos \theta.$$

Векторы называют *ортгоналичными* и пишут $\mathbf{a} \perp \mathbf{b}$, если $\mathbf{a} \cdot \mathbf{b} = 0$.

Для векторов справедливо неравенство Коши – Буняковского – Шварца

$$|\mathbf{a} \cdot \mathbf{b}| \leq |\mathbf{a}| \cdot |\mathbf{b}|.$$

Здесь равенство возможно только при $\mathbf{a} = \alpha \mathbf{b}$.

Евклидова норма также называется l_2 -нормой и обозначается $\|\mathbf{a}\|_2$. В векторном анализе применяются l_1 -норма и l_∞ -норма:

$$\|\mathbf{a}\|_1 = \sum_{i=1}^n |a_i|, \quad \|\mathbf{a}\|_\infty = \max_{1 \leq i \leq n} |a_i|.$$

Представленные нормы векторов – частные случаи l_p -нормы

$$\|\mathbf{a}\|_p = \left(\sum_{i=1}^n |a_i|^p \right)^{1/p}.$$

Любая норма вектора обладает свойствами:

- 1) $\|\mathbf{a}\| \geq 0 \quad \forall \mathbf{a} \in R^n$, $\|\mathbf{a}\| = 0$ тогда и только тогда, когда $\mathbf{a} = \mathbf{0}$;
- 2) $\|\alpha \mathbf{a}\| = |\alpha| \|\mathbf{a}\| \quad \forall \mathbf{a} \in R^n, \forall \alpha \in R$;
- 3) $\|\mathbf{a} + \mathbf{b}\| \leq \|\mathbf{a}\| + \|\mathbf{b}\|, \quad \forall \mathbf{a}, \mathbf{b} \in R^n$ – неравенство треугольника.

Две нормы $\|\cdot\|_r$ и $\|\cdot\|_s$ называются *эквивалентными*, если существуют $\alpha, \beta > 0$ такие, что

$$\alpha \|\mathbf{a}\|_r \leq \|\mathbf{a}\|_s \leq \beta \|\mathbf{a}\|_r \quad \forall \mathbf{a} \in R^n.$$

В частности,

$$\|\mathbf{a}\|_2 \leq \|\mathbf{a}\|_1 \leq \sqrt{n} \|\mathbf{a}\|_2, \quad \|\mathbf{a}\|_\infty \leq \|\mathbf{a}\|_2 \leq \sqrt{n} \|\mathbf{a}\|_\infty,$$

$$\|\mathbf{a}\|_\infty \leq \|\mathbf{a}\|_1 \leq n \|\mathbf{a}\|_\infty, \quad \|\mathbf{a}\|_\infty \leq \|\mathbf{a}\|_2 \leq \|\mathbf{a}\|_1.$$

Таким образом, нормы $\|\mathbf{a}\|_1$, $\|\mathbf{a}\|_2$ и $\|\mathbf{a}\|_\infty$ являются эквивалентными.

Расстоянием между двумя точками $\mathbf{x} = (x_1, x_2, \dots, x_n)$ и $\mathbf{y} = (y_1, y_2, \dots, y_n)$ пространства R^n называется евклидова норма разностей их координат

$$\|\mathbf{x} - \mathbf{y}\| = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2}.$$

Расстояние обладает свойствами:

- 1) $\|\mathbf{x} - \mathbf{y}\| \geq 0$, $\|\mathbf{x} - \mathbf{y}\| = 0$ тогда и только тогда, когда $\mathbf{x} = \mathbf{y}$;
- 2) $\|\mathbf{x} - \mathbf{y}\| = \|\mathbf{y} - \mathbf{x}\|$;
- 3) $\|\mathbf{x} - \mathbf{z}\| = \|\mathbf{x} - \mathbf{y}\| + \|\mathbf{y} - \mathbf{z}\|$ – неравенство треугольника.

Последовательность векторов $\{\mathbf{x}_k\}$ называется *сходящейся* к вектору \mathbf{x}^* , если

$$\lim_{k \rightarrow \infty} \|\mathbf{x}_k - \mathbf{x}^*\| = 0.$$

Понятие сходимости последовательности векторов положено в основу исследования скорости сходимости методов многомерной оптимизации.

5.3. Матрицы и действия с ними

Матрица размера $m \times n$ представляет собой прямоугольную таблицу действительных чисел, состоящую из m строк и n столбцов

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix},$$

где a_{ij} – действительные числа, называемые *элементами* матрицы. Элементы a_{ij} при $i = j$ называются *диагональными* элементами, если же $i \neq j$, то a_{ij} называются *недиагональными* элементами. Диагональные элементы составляют *главную диагональ* матрицы.

Элементы каждого столбца матрицы составляют вектор, который

называется *вектор-столбцом*. Аналогично каждая строка матрицы определяет *вектор-строку*. Вектор можно рассматривать как матрицу специального вида, которая содержит либо только одну строку, либо лишь один столбец. Матрица с одинаковым количеством строк и столбцов называется *квадратной матрицей*.

Если заменить строки матрицы \mathbf{A} столбцами, то получится *транспонированная* к \mathbf{A} матрица, которая обозначается \mathbf{A}^T . В частности, если \mathbf{a} – вектор-строка, то \mathbf{a}^T – вектор-столбец.

Квадратная матрица \mathbf{A} называется *симметрической*, если для ее элементов $a_{ij} = a_{ji}$, то есть симметричные относительно главной диагонали элементы одинаковы. Для симметрической матрицы $\mathbf{A}^T = \mathbf{A}$.

Две матрицы называются *равными*, если они имеют одинаковые размеры и их соответствующие элементы совпадают.

Квадратная матрица, все диагональные элементы которой равны 1, а внедиагональные элементы – нулю, называется *единичной* матрицей и обозначается через \mathbf{E} . Матрица, которая содержит только равные нулю элементы, называется *нулевой* матрицей и обозначается \mathbf{O} .

Рангом матрицы называется максимальное число ее линейно независимых строк или столбцов.

Сумма или *разность* двух матриц \mathbf{A} и \mathbf{B} одинакового размера есть матрица того же размера \mathbf{C} , элементы которой вычисляются по формулам $c_{ij} = a_{ij} + b_{ij}$ или $c_{ij} = a_{ij} - b_{ij}$. При этом пишут $\mathbf{C} = \mathbf{A} + \mathbf{B}$ или $\mathbf{C} = \mathbf{A} - \mathbf{B}$.

Скалярное произведение двух вектор-столбцов \mathbf{a} и \mathbf{b} представимо в виде $\mathbf{a}^T \mathbf{b} = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$. При этом $\mathbf{a}^T \mathbf{b} = \mathbf{b}^T \mathbf{a}$. Условие ортогональности векторов $\mathbf{a} \perp \mathbf{b}$ имеет вид $\mathbf{a}^T \mathbf{b} = 0$. Неравенство Коши-Буняковского-Шварца можно представить в виде $|\mathbf{a}^T \mathbf{b}| \leq \|\mathbf{a}\| \cdot \|\mathbf{b}\|$.

Произведение \mathbf{AB} двух матриц \mathbf{A} и \mathbf{B} определено тогда и только тогда, когда размеры матриц согласованы, то есть количество столбцов матрицы \mathbf{A} равно количеству строк матрицы \mathbf{B} . Если \mathbf{A} –

матрица размера $m \times n$, а \mathbf{B} – матрица размера $n \times p$, то произведение этих матриц $\mathbf{AB} = \mathbf{C}$ представляет собой матрицу размера $m \times p$. Элемент матрицы \mathbf{C} , расположенный на пересечении i -той строки и j -того столбца, определяется по формуле

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \dots + a_{in}b_{nj}.$$

Это означает, что необходимо найти скалярное произведение i -той строки матрицы \mathbf{A} и j -того столбца матрицы \mathbf{B} , то есть произведение матриц выполняется по правилу «строка на столбец». В частности, всегда можно перемножать квадратные матрицы одинакового размера. В общем случае $\mathbf{AB} \neq \mathbf{BA}$.

Произведением матрицы \mathbf{A} на некоторое число α называется матрица $\alpha\mathbf{A}$ с элементами αa_{ij} .

Определителем квадратной матрицы \mathbf{A} называется число, обозначаемое $\det \mathbf{A}$ и получаемое с помощью арифметических операций над элементами \mathbf{A} . Если \mathbf{A} – матрица размера 2×2 , то

$$\det \mathbf{A} = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}.$$

Если \mathbf{A} – матрица размера $n \times n$, то

$$\det \mathbf{A} = \sum_{i=1}^n a_{i1}(-1)^{i+1}M_{i1},$$

где M_{i1} называется *минором* элемента a_{i1} и представляет собой определитель подматрицы матрицы \mathbf{A} , полученной путем исключения строки i и столбца 1.

Матрица называется *вырожденной* или *особенной*, если ее определитель равен нулю. Если же определитель не равен нулю, то соответствующая ему матрица называется *невырожденной* или *неособенной*. При транспонировании матрицы ее определитель не изменяется.

Матрица, обратная к невырожденной квадратной матрице \mathbf{A} , обозначается как \mathbf{A}^{-1} и представляет собой невырожденную квадратную матрицу, обладающую свойством $\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{E}$. Определитель обратной матрицы $\det \mathbf{A}^{-1} = 1/\det \mathbf{A}$.

Операции над матрицами обладают свойствами:

- 1) $\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$;
- 2) $(\mathbf{A} + \mathbf{B}) + \mathbf{C} = \mathbf{A} + (\mathbf{B} + \mathbf{C})$;
- 3) $\alpha(\mathbf{A} + \mathbf{B}) = \alpha\mathbf{A} + \alpha\mathbf{B}$;
- 4) $(\alpha + \beta)\mathbf{A} = \alpha\mathbf{A} + \beta\mathbf{A}$;
- 5) $\mathbf{A}\mathbf{E} = \mathbf{A}$, $\mathbf{E}\mathbf{A} = \mathbf{A}$;
- 6) $\alpha(\mathbf{A}\mathbf{B}) = (\alpha\mathbf{A})\mathbf{B} = \mathbf{A}(\alpha\mathbf{B})$;
- 7) $(\mathbf{A} + \mathbf{B})\mathbf{C} = \mathbf{A}\mathbf{C} + \mathbf{B}\mathbf{C}$, $\mathbf{A}(\mathbf{B} + \mathbf{C}) = \mathbf{A}\mathbf{B} + \mathbf{A}\mathbf{C}$;
- 8) $(\mathbf{A}\mathbf{B})\mathbf{C} = \mathbf{A}(\mathbf{B}\mathbf{C})$;
- 9) $(\mathbf{A} + \mathbf{B})^T = \mathbf{A}^T + \mathbf{B}^T$;
- 10) $(\mathbf{A}\mathbf{B})^T = \mathbf{B}^T\mathbf{A}^T$;
- 11) $\det(\mathbf{A}\mathbf{B}) = \det \mathbf{A} \det \mathbf{B}$;
- 12) $(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$.

Если для квадратной матрицы \mathbf{A} размера n и ненулевого вектор-столбца $\mathbf{x} \in R^n$, $\mathbf{x} \neq \mathbf{0}$ выполняется равенство $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$, то число λ называется собственным значением, а вектор \mathbf{x} называется собственным вектором матрицы \mathbf{A} .

В соответствии с l_p -нормой вектора вводится l_p -норма матрицы

$$\|\mathbf{A}\|_p = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{A}\mathbf{x}\|_p}{\|\mathbf{x}\|_p} = \max_{\|\mathbf{x}\|_p=1} \|\mathbf{A}\mathbf{x}\|_p,$$

о которой говорят, что она индуцирована векторной l_p -нормой. В частности, определены максимальная столбцовая норма, максимальная строчная норма и спектральная норма:

$$\|\mathbf{A}\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^m |a_{ij}|, \quad \|\mathbf{A}\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|, \quad \|\mathbf{A}\|_2 = [\lambda_{\max}(\mathbf{A}^T \mathbf{A})]^{1/2}.$$

Спектральная норма матрицы \mathbf{A} равна квадратному корню из максимального собственного значения матрицы $\mathbf{A}^T \mathbf{A}$.

Последовательность матриц $\{\mathbf{A}_k\}$ называется сходящейся к матрице \mathbf{A} , если

$$\lim_{k \rightarrow \infty} \|\mathbf{A}_k - \mathbf{A}\| = 0.$$

5.4. Формула Шермана – Моррисона и ее применение

Пусть \mathbf{A} – невырожденная квадратная матрица размера n , а \mathbf{a} и \mathbf{b} – произвольные n -мерные векторы столбцы. Если $1 + \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a} \neq 0$, то матрица $\mathbf{A} + \mathbf{a} \mathbf{b}^T \neq \mathbf{0}$ невырожденная и имеет обратную матрицу

$$(\mathbf{A} + \mathbf{a} \mathbf{b}^T)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T \mathbf{A}^{-1}}{1 + \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a}}. \quad (5.1)$$

Это равенство называется *формулой Шермана – Моррисона*. Для доказательства ее истинности умножим равенство (5.1) на матрицу $\mathbf{A} + \mathbf{a} \mathbf{b}^T$:

$$\begin{aligned} (\mathbf{A} + \mathbf{a} \mathbf{b}^T)^{-1} (\mathbf{A} + \mathbf{a} \mathbf{b}^T) &= \mathbf{E} + \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T - \frac{\mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T \mathbf{A}^{-1} \mathbf{A} + \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T}{1 + \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a}} = \\ &= \mathbf{E} + \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T - \frac{\mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T + (\mathbf{b}^T \mathbf{A}^{-1} \mathbf{a}) \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T}{1 + \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a}} = \\ &= \mathbf{E} + \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T - \mathbf{A}^{-1} \mathbf{a} \mathbf{b}^T \frac{1 + \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a}}{1 + \mathbf{b}^T \mathbf{A}^{-1} \mathbf{a}} = \mathbf{E}. \end{aligned}$$

Применим формулу Шермана – Моррисона для вывода модифи-

цированной формулы Бройдена – Флетчера – Гольдфарба – Шанно (4.35) обращением матрицы (4.29)

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{\mathbf{p}_k \mathbf{p}_k^T}{\mathbf{p}_k^T \mathbf{s}_k} - \frac{\mathbf{H}_k \mathbf{s}_k (\mathbf{H}_k \mathbf{s}_k)^T}{(\mathbf{H}_k \mathbf{s}_k)^T \mathbf{s}_k}, \quad (5.2)$$

Введем обозначения:

$$\mathbf{G}_{k+1} = \mathbf{H}_{k+1}^{-1}, \quad \mathbf{G}_k = \mathbf{H}_k^{-1}, \quad (5.3)$$

$$\mathbf{X} = \mathbf{G}_{k+1}, \quad \mathbf{Y} = \mathbf{H}_{k+1}, \quad \mathbf{H} = \mathbf{H}_k, \quad \mathbf{G} = \mathbf{G}_k, \quad \mathbf{p} = \mathbf{p}_k, \quad \mathbf{s} = \mathbf{s}_k, \quad (5.4)$$

$$\mathbf{u} = \mathbf{G}\mathbf{p}, \quad \mathbf{v} = \mathbf{H}\mathbf{s}, \quad \alpha = (\mathbf{p}^T \mathbf{s})^{-1}, \quad \beta = -(\mathbf{v}^T \mathbf{s})^{-1}. \quad (5.5)$$

Формула (5.2) примет вид:

$$\mathbf{Y} = \mathbf{H} + \alpha \mathbf{p} \mathbf{p}^T + \beta \mathbf{v} \mathbf{v}^T. \quad (5.6)$$

Рассмотрим матрицу $\mathbf{V} = \mathbf{H} + \alpha \mathbf{p} \mathbf{p}^T$. По формуле (5.1) имеем:

$$\mathbf{U} = \mathbf{V}^{-1} = (\mathbf{H} + \alpha \mathbf{p} \mathbf{p}^T)^{-1} = \mathbf{H}^{-1} - \frac{\mathbf{H}^{-1} \alpha \mathbf{p} \mathbf{p}^T \mathbf{H}^{-1}}{1 + \mathbf{p}^T \mathbf{H}^{-1} \alpha \mathbf{p}}.$$

С учетом обозначений (5.3)–(5.5)

$$\mathbf{U} = \mathbf{G} - \frac{\alpha \mathbf{G} \mathbf{p} \mathbf{p}^T \mathbf{G}}{1 + \alpha \mathbf{p}^T \mathbf{G} \mathbf{p}} = \mathbf{G} - \frac{\mathbf{u} \mathbf{u}^T}{\mathbf{p}^T \mathbf{s} + \mathbf{p}^T \mathbf{u}}.$$

Обозначая

$$\gamma = (\mathbf{p}^T \mathbf{s} + \mathbf{p}^T \mathbf{u})^{-1}, \quad (5.7)$$

получим:

$$\mathbf{U} = \mathbf{V}^{-1} = \mathbf{G} - \gamma \mathbf{u} \mathbf{u}^T. \quad (5.8)$$

Рассмотрим матрицу (5.6) $\mathbf{Y} = \mathbf{V} + \beta \mathbf{v} \mathbf{v}^T$. По формуле (5.1) имеем:

$$\mathbf{X} = \mathbf{Y}^{-1} = (\mathbf{V} + \beta \mathbf{v} \mathbf{v}^T)^{-1} = \mathbf{V}^{-1} - \frac{\mathbf{V}^{-1} \beta \mathbf{v} \mathbf{v}^T \mathbf{V}^{-1}}{1 + \mathbf{v}^T \mathbf{V}^{-1} \beta \mathbf{v}}.$$

Учитывая (5.8),

$$\mathbf{X} = \mathbf{U} - \frac{\beta \mathbf{U} \mathbf{v} \mathbf{v}^T \mathbf{U}}{1 + \beta \mathbf{v}^T \mathbf{U} \mathbf{v}}.$$

Перепишем это равенство в виде:

$$\mathbf{X} = \mathbf{U} - \mathbf{N}/D, \quad (5.9)$$

где

$$\mathbf{N} = \mathbf{U} \mathbf{v} \mathbf{v}^T \mathbf{U}, \quad D = \beta^{-1} (1 + \beta \mathbf{v}^T \mathbf{U} \mathbf{v}). \quad (5.10)$$

На основании (5.8) имеем:

$$\begin{aligned} \mathbf{N} &= (\mathbf{G} - \gamma \mathbf{u} \mathbf{u}^T) \mathbf{v} \mathbf{v}^T (\mathbf{G} - \gamma \mathbf{u} \mathbf{u}^T), \\ \mathbf{N} &= \mathbf{G} \mathbf{v} \mathbf{v}^T \mathbf{G} - \gamma \mathbf{G} \mathbf{v} \mathbf{v}^T \mathbf{u} \mathbf{u}^T - \gamma \mathbf{u} \mathbf{u}^T \mathbf{v} \mathbf{v}^T \mathbf{G} + \gamma^2 \mathbf{u} \mathbf{u}^T \mathbf{v} \mathbf{v}^T \mathbf{u} \mathbf{u}^T. \end{aligned} \quad (5.11)$$

С учетом обозначений (5.4) и (5.5)

$$\mathbf{v} = \mathbf{H} \mathbf{s} = \mathbf{G}^{-1} \mathbf{s}, \quad \mathbf{G} \mathbf{v} = \mathbf{G} \mathbf{G}^{-1} \mathbf{s} = \mathbf{E} \mathbf{s} = \mathbf{s}.$$

Поскольку матрицы \mathbf{H} и \mathbf{G} симметрические, $\mathbf{H}^T = \mathbf{H}$, $\mathbf{G}^T = \mathbf{G}$, то

$$\mathbf{v}^T \mathbf{G} = (\mathbf{G} \mathbf{v})^T = \mathbf{s}^T, \quad \mathbf{v}^T \mathbf{u} = \mathbf{u}^T \mathbf{v} = \mathbf{s}^T \mathbf{H} \mathbf{G} \mathbf{p} = \mathbf{s}^T \mathbf{E} \mathbf{p} = \mathbf{s}^T \mathbf{p}.$$

Поэтому выражение (5.11) примет вид:

$$\mathbf{N} = \mathbf{s} \mathbf{s}^T - \gamma \mathbf{s}^T \mathbf{p} (\mathbf{s} \mathbf{u}^T + \mathbf{u} \mathbf{s}^T) + \gamma^2 (\mathbf{s}^T \mathbf{p})^2 \mathbf{u} \mathbf{u}^T. \quad (5.12)$$

На основании (5.5) и (5.8) для D в (5.10) получим:

$$D = \beta^{-1} (1 + \beta \mathbf{v}^T \mathbf{U} \mathbf{v}) = -\mathbf{v}^T \mathbf{s} + \mathbf{v}^T (\mathbf{G} - \gamma \mathbf{u} \mathbf{u}^T) \mathbf{v} = \mathbf{v}^T (-\mathbf{s} + \mathbf{G} \mathbf{v} - \gamma \mathbf{u} \mathbf{u}^T \mathbf{v}),$$

$$D = \mathbf{v}^T (-\mathbf{s} + \mathbf{s} - \gamma \mathbf{u} \mathbf{s}^T \mathbf{p}), \quad D = -\gamma (\mathbf{s}^T \mathbf{p})^2.$$

Подставим последнее выражение, выражения (5.8) и (5.12) в (5.9):

$$\mathbf{X} = \mathbf{G} - \gamma \mathbf{u} \mathbf{u}^T + \frac{\mathbf{s} \mathbf{s}^T - \gamma \mathbf{s}^T \mathbf{p} (\mathbf{s} \mathbf{u}^T + \mathbf{u} \mathbf{s}^T) + \gamma^2 (\mathbf{s}^T \mathbf{p})^2 \mathbf{u} \mathbf{u}^T}{\gamma (\mathbf{s}^T \mathbf{p})^2},$$

$$\mathbf{X} = \mathbf{G} - \gamma \mathbf{u} \mathbf{u}^T + \frac{\mathbf{s} \mathbf{s}^T}{\gamma (\mathbf{s}^T \mathbf{p})^2} - \frac{\mathbf{s} \mathbf{u}^T + \mathbf{u} \mathbf{s}^T}{\gamma \mathbf{s}^T \mathbf{p}} + \gamma \mathbf{u} \mathbf{u}^T.$$

С учетом (5.7) получим:

$$\mathbf{X} = \mathbf{G} + (\mathbf{p}^T \mathbf{s} + \mathbf{p}^T \mathbf{u}) \frac{\mathbf{s} \mathbf{s}^T}{(\mathbf{s}^T \mathbf{p})^2} - \frac{\mathbf{s} \mathbf{u}^T + \mathbf{u} \mathbf{s}^T}{\mathbf{s}^T \mathbf{p}},$$

$$\mathbf{X} = \mathbf{G} + \left(1 + \frac{\mathbf{p}^T \mathbf{u}}{\mathbf{s}^T \mathbf{p}} \right) \frac{\mathbf{s} \mathbf{s}^T}{\mathbf{s}^T \mathbf{p}} - \frac{\mathbf{s} \mathbf{u}^T + \mathbf{u} \mathbf{s}^T}{\mathbf{s}^T \mathbf{p}}.$$

Возвращаясь к исходным обозначениям по равенствам (5.3)–(5.5), получим окончательно модифицированную формулу БФГШ (4.35)

$$\mathbf{G}_{k+1} = \mathbf{G}_k + \left(1 + \frac{\mathbf{p}_k^T \mathbf{G}_k \mathbf{p}_k}{\mathbf{s}_k^T \mathbf{p}_k} \right) \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{p}_k} - \frac{\mathbf{s}_k (\mathbf{G}_k \mathbf{p}_k)^T + \mathbf{G}_k \mathbf{p}_k \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{p}_k}.$$

5.5. Дифференцирование функций многих переменных

Пусть функция $f(\mathbf{x})$ векторного аргумента $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ задана в некоторой области $D \subset R^n$ и пусть \mathbf{x}_0 – некоторая точка этой области. Функция $f(\mathbf{x})$ называется *непрерывной* в точке \mathbf{x}_0 , если для любого числа $\varepsilon > 0$ существует такое число $\delta > 0$, что из неравенства $\|\mathbf{x} - \mathbf{x}_0\| < \delta$ следует неравенство $|f(\mathbf{x}) - f(\mathbf{x}_0)| < \varepsilon$.

Пусть функция $f(\mathbf{x})$ векторного аргумента $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ задана в некоторой области $D \subset R^n$ и пусть \mathbf{x}_0 – произвольная точка этой области. Придадим переменной x_i приращение δ такое, чтобы было $\mathbf{x}_0 + \delta \mathbf{e}_i \in D$, где $\mathbf{e}_i = (0, 0, \dots, 0, 1, 0, \dots, 0)^T$ – орт оси Ox_i , представляющий собой вектор с i -той единичной проекцией и остальными нулевыми проекциями. Вычислим разность $\Delta_i f = f(\mathbf{x}_0 + \delta \mathbf{e}_i) - f(\mathbf{x}_0)$

и составим отношение $\Delta_i f / \delta$. Если существует предел

$$\lim_{\delta \rightarrow 0} \Delta_i f / \delta ,$$

то он называется *частной производной* функции $f(\mathbf{x})$ по переменной x_i в точке \mathbf{x}_0 и обозначается $\partial f(\mathbf{x}_0) / \partial x_i$.

Итак, по определению

$$\frac{\partial f(\mathbf{x}_0)}{\partial x_i} = \lim_{\delta \rightarrow 0} \frac{f(\mathbf{x}_0 + \delta \mathbf{e}_i) - f(\mathbf{x}_0)}{\delta} .$$

Частные производные вычисляются по обычным правилам и формулам дифференцирования, причем при вычислении $\partial f / \partial x_i$ дифференцирование ведется по переменной x_i , а остальные переменные считаются неизменными.

Частные производные функции двух переменных $f(x_1, x_2)$ геометрически представляют собой тангенсы углов наклона касательных к сечениям графика этой функции плоскостями $x_1 = C$ и $x_2 = C$.

Если для функции $f(\mathbf{x})$ в точке \mathbf{x}_0 существуют все частные производные первого порядка, то она называется *дифференцируемой* в точке \mathbf{x}_0 . Если функция $f(\mathbf{x})$ дифференцируема в точке \mathbf{x}_0 , то она и непрерывна в этой точке.

Пусть функция $f(\mathbf{x})$ является дифференцируемой при всех $\mathbf{x} \in R^n$. Тогда она называется *гладкой*.

Вектор-столбец частных производных функции $f(\mathbf{x})$ в точке \mathbf{x}_0 называется *градиентом* и обозначается

$$\nabla f(\mathbf{x}_0) = \left(\frac{\partial f(\mathbf{x}_0)}{\partial x_1}, \frac{\partial f(\mathbf{x}_0)}{\partial x_2}, \dots, \frac{\partial f(\mathbf{x}_0)}{\partial x_n} \right)^T .$$

Символ «набла» ∇ служит для обозначения оператора градиента

$\nabla = (\partial/\partial x_1, \partial/\partial x_2, \dots, \partial/\partial x_n)^T$. Градиент также обозначается через $\mathbf{g}(\mathbf{x})$. Вектор градиента определяет направление наиболее быстрого возрастания функции в данной точке.

Поскольку по определению градиента $\nabla x_i = \mathbf{e}_i$, то

$$\nabla \mathbf{x}^T = \nabla(x_1, x_2, \dots, x_n) = (\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n) = \mathbf{E}.$$

Если векторы-столбцы \mathbf{u} и \mathbf{v} зависят от \mathbf{x} , то по правилу производной произведения функций

$$\nabla(\mathbf{u}^T \mathbf{v}) = (\nabla \mathbf{u}^T) \mathbf{v} + (\nabla \mathbf{v}^T) \mathbf{u}.$$

Пусть функция $f(\mathbf{x})$ имеет частные производные $\partial f / \partial x_i$, $i = \overline{1, n}$. Предположим, что эти частные производные в свою очередь дифференцируемы по всем переменным. Тогда, дифференцируя $\partial f / \partial x_i$ по x_i , получим *частную производную второго порядка* функции $f(\mathbf{x})$ по переменной x_i . Она обозначается $\partial^2 f / \partial x_i^2$.

Дифференцируя $\partial f / \partial x_i$ по другой переменной x_j , получим *смешанную частную производную второго порядка*, которая обозначается

$$\frac{\partial^2 f}{\partial x_j \partial x_i}.$$

Теорема Шварца. Пусть функция $f(\mathbf{x})$ имеет в точке \mathbf{x}_0 и в некоторой ее окрестности частные производные

$$\frac{\partial f}{\partial x_i}, \quad \frac{\partial f}{\partial x_j}, \quad \frac{\partial^2 f}{\partial x_j \partial x_i}, \quad \frac{\partial^2 f}{\partial x_i \partial x_j}.$$

Тогда

$$\frac{\partial^2 f(\mathbf{x}_0)}{\partial x_j \partial x_i} = \frac{\partial^2 f(\mathbf{x}_0)}{\partial x_i \partial x_j}.$$

Пусть функция $f(\mathbf{x})$ является дважды дифференцируемой по всем переменным. Тогда для такой функции можно составить матрицу вторых частных производных, которая называется *матрицей Гессе*

$$\nabla^2 f(\mathbf{x}) = \begin{pmatrix} \frac{\partial^2 f(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_2 \partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(\mathbf{x})}{\partial x_n^2} \end{pmatrix}.$$

В силу теоремы Шварца

$$\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\mathbf{x})}{\partial x_j \partial x_i}, \quad i, j = \overline{1, n}, \quad i \neq j.$$

Поэтому матрица Гессе является симметрической матрицей и для нее

$$\nabla^2 f(\mathbf{x})^T = \nabla^2 f(\mathbf{x}).$$

Матрица Гессе также обозначается как $\mathbf{H}(\mathbf{x})$.

5.6. Формула Тейлора

Пусть функция $f(x)$ скалярного действительного аргумента $x \in R$ в любой точке x_0 n раз дифференцируема. Тогда для нее справедлива формула Тейлора

$$f(x+h) = f(x) + f'(x)h + \frac{1}{2!} f''(x)h^2 + \dots + \frac{1}{n!} f^{(n)}(x)h^n + o(h^n),$$

где h – приращение аргумента, $o(h^n)$ – бесконечно малая более высокого порядка, чем h^n , называемая остаточным членом формулы

Тейлора в форме Пеано.

Пусть теперь функция $f(\mathbf{x})$ векторного аргумента $\mathbf{x} \in R^n$ в любой точке дважды дифференцируема. Тогда для нее формула Тейлора

$$f(\mathbf{x} + \mathbf{h}) = f(\mathbf{x}) + \sum_{i=1}^n \frac{\partial f(\mathbf{x})}{\partial x_i} h_i + \frac{1}{2!} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} h_i h_j + o(\|\mathbf{h}\|^2),$$

где $\mathbf{h} = (h_1, h_2, \dots, h_n)^T$ – векторное приращение аргумента. Эту формулу можно представить в векторном виде

$$f(\mathbf{x} + \mathbf{h}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \mathbf{h} + \frac{1}{2!} \mathbf{h}^T \nabla^2 f(\mathbf{x}) \mathbf{h} + o(\|\mathbf{h}\|^2),$$

где $\nabla f(\mathbf{x})$ и $\nabla^2 f(\mathbf{x})$ – градиент и матрица Гессе функции $f(\mathbf{x})$ соответственно.

5.7. Квадратичные формы

Функция n переменных $f(x_1, x_2, \dots, x_n)$ называется *квадратичной формой*, если

$$f(x_1, x_2, \dots, x_n) = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j.$$

Вводя обозначения для вектора переменных $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ и квадратной матрицы \mathbf{A} с элементами a_{ij} , квадратичную форму можно представить в виде

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x}.$$

Без потери общности матрицу \mathbf{A} можно всегда предполагать симметрической. В противном случае \mathbf{A} следует заменить симметрической матрицей $(\mathbf{A} + \mathbf{A}^T)/2$, значения квадратичной формы при этом не изменятся.

Симметрическая матрица \mathbf{A} называется *неотрицательно определенной*, если $\mathbf{x}^T \mathbf{A} \mathbf{x} \geq 0$ для всех $\mathbf{x} \in R^n$. При этом пишут $\mathbf{A} \geq 0$.

Симметрическая матрица \mathbf{A} называется *положительно определенной*, если $\mathbf{x}^T \mathbf{A} \mathbf{x} > 0$ для всех $\mathbf{x} \neq 0$. При этом пишут $\mathbf{A} > 0$.

Симметрическая матрица \mathbf{A} называется *неположительно определенной*, если $\mathbf{x}^T \mathbf{A} \mathbf{x} \leq 0$ для всех $\mathbf{x} \in R^n$. При этом пишут $\mathbf{A} \leq 0$. Матрица \mathbf{A} неположительно определенная тогда и только тогда, когда $-\mathbf{A}$ есть неотрицательно определенная матрица.

Симметрическая матрица \mathbf{A} называется *отрицательно определенной*, если $\mathbf{x}^T \mathbf{A} \mathbf{x} < 0$ для всех $\mathbf{x} \neq 0$. При этом пишут $\mathbf{A} < 0$. Матрица \mathbf{A} отрицательно определенная тогда и только тогда, когда $-\mathbf{A}$ – положительно определенная матрица.

Симметрическая матрица \mathbf{A} называется *неопределенной*, если квадратичная форма $\mathbf{x}^T \mathbf{A} \mathbf{x}$ может принимать как положительные, так и отрицательные значения.

Для установления положительной определенности матрицы \mathbf{A} составим из ее элементов определители

$$\Delta_1 = a_{11}, \quad \Delta_2 = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}, \quad \Delta_3 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}, \quad \dots,$$

$$\Delta_n = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}.$$

Очевидно, $\det \mathbf{A} = \Delta_n$. Определители $\Delta_1, \Delta_2, \Delta_3, \dots, \Delta_{n-1}$ называются *главными диагональными минорами* матрицы \mathbf{A} .

Критерий Сильвестра – Якоби. Для того, чтобы квадратная симметрическая матрица была положительно определенной,

необходимо и достаточно, чтобы все ее главные диагональные миноры, включая и определитель матрицы, были положительными.

Таким образом, если для матрицы \mathbf{A} имеем $\Delta_1 > 0$, $\Delta_2 > 0$, $\Delta_3 > 0$, ..., $\Delta_{n-1} > 0$, $\Delta_n > 0$, то \mathbf{A} – положительно определенная матрица. Все диагональные элементы положительно определенной матрицы должны быть положительными.

Чтобы установить, что матрица является отрицательно определенной, следует умножить ее на -1 и проверить полученную матрицу на положительную определенность. Тогда из критерия Сильвестра-Якоби следует, что для того, чтобы матрица была отрицательно определенной, необходимо и достаточно, чтобы знаки ее главных диагональных миноров, включая и определитель этой матрицы, чередовались, начиная со знака « $-$ ». То есть, если $\Delta_1 < 0$, $\Delta_2 > 0$, $\Delta_3 < 0$, ..., $(-1)^n \Delta_n > 0$, то \mathbf{A} – отрицательно определенная матрица. Все диагональные элементы отрицательно определенной матрицы должны быть отрицательными.

При проверке матрицы на неотрицательную определенность достаточно установить, что все ее диагональные элементы и все главные диагональные миноры, включая определитель матрицы, неотрицательны.

При проверке матрицы на неположительную определенность достаточно установить, что все ее диагональные элементы и все главные диагональные миноры, включая определитель матрицы, неположительные.

При проверке матрицы на неопределенность достаточно убедиться в том, что, по крайней мере, два из ее диагональных элементов имеют разные знаки.

СПИСОК ЛИТЕРАТУРЫ

1. Аттетков А.В. Методы оптимизации : учебник для вузов / А. В. Аттетков, С.В. Галкин, В.С Зарубин ; под ред. В.С. Зарубина, А.П. Крищенко. – М. : Изд-во МГТУ им. Н.Э. Баумана, 2003. – 440 с.
2. Банди Б. Методы оптимизации. Вводный курс / Б. Банди ; пер. с англ. О.В. Шихеевой. – М. : Радио и связь, 1988. – 128 с.
3. Бейко И.В. Методы и алгоритмы решения задач оптимизации / И.В. Бейко, Б.Н. Бублик, П.Н. Зинько. – К. : Вища школа, 1983. – 512 с.
4. Воробьев Н.Н. Числа Фибоначчи / Н.Н. Воробьев. – 4-е изд., доп. – М. : Наука. Гл. ред. физ.-мат. лит., 1978. – 142 с.
5. Гилл Ф. Практическая оптимизация / Ф. Гилл, У. Мюррей, М. Райт ; пер. с англ. В.Ю. Лебедева. – М. : Мир, 1985. – 512 с.
6. Дэннис Д. Численные методы безусловной оптимизации и решения нелинейных уравнений / Д. Дэннис, Р. Шнабель ; пер. с англ. О.П. Бурдакова – М. : Мир, 1988. – 440 с.
7. Измаилов А.Ф. Численные методы оптимизации : учеб. пособ. / А.Ф. Измаилов, М.В. Солодов. – М. : ФИЗМАТЛИТ, 2005. – 304 с.
8. Карманов В.Г. Математическое программирование / В.Г. Карманов. – 2-е изд., перераб. и доп. – М. : Наука. Гл. ред. физ.-мат. лит., 1980. – 256 с.
9. Короп В.Ф. Принцип равномерного вклада и регулирование шагов покоординатного случайного поиска / В.Ф. Короп // Численные методы нелинейного программирования. – К. : Наук. думка. – 1976. – С. 33–41.
10. Мину М. Математическое программирование. Теория и алгоритмы / М. Мину ; пер. с фр. А.И. Штерна. – М. : Наука. Гл. ред. физ.-мат. лит., 1990. – 488 с.
11. Полак Д. Численные методы оптимизации. Единый подход / Д. Полак ; пер. с англ. Ф.И. Ерешко. – М. : Мир, 1974. – 376 с.
12. Пантелеев А. В. Методы оптимизации в примерах и задачах :

учеб. пособ. / А. В. Пантелеев, Т.А. Летова. – М. : Высш. шк., 2002. – 544 с.

13. Раскин Л.Г. Математическое программирование : учебно-метод. пособ. / Л.Г. Раскин. – Харьков : ПП «ГЕММА», 1986. – 68 с.

14. Реклейтис Г. Оптимизация в технике : в 2 кн. / Г. Реклейтис, А. Рейвиндран, К. Рэгсдел. Кн. 1 ; пер. с англ. В.Я. Алтаева, В.И. Моторина. – М. : Мир, 1986. – 349 с.

15. Северин В.П. Методы одномерного поиска : учебно-метод. пособ. по курсу «Методы оптимизации» / В.П. Северин. – Х. : НТУ «ХПИ», 2012. – 112 с.

16. Струченков В.И. Методы оптимизации. Основы теории, задачи, обучающие компьютерные программы : учеб. пособ. / В.И. Струченков. – М. : Экзамен, 2005. – 256 с.

17. Сухарев А.Г. Курс методов оптимизации : учеб. пособ. / А.Г. Сухарев, А.В. Тимохов, В.В. Федоров. – 2-е изд. – М. : ФИЗМАТ-ЛИТ, 2005. – 368 с.

18. Форсайт Д. Машинные методы математических вычислений / Д. Форсайт, М. Малькольм, К. Моулер ; пер. с англ. Х.Д. Икрамова. – М. : Мир, 1980. – 280 с.

19. Химмельблау Д. Прикладное нелинейное программирование / Д. Химмельблау ; пер. с англ. И.М. Быховской, Б.Т. Вавилова. – М. : Мир, 1975. – 536 с.

20. Fletcher R. Practical methods of optimization / R. Fletcher. – 2-d edition. – UK, Chichester : John Wiley & Sons Ltd, 2000. – 436 p.

21. Luenberger D. G. Linear and nonlinear programming / D.G. Luenberger, Y. Yinyu. – 3-d edition. – USA, New York : Springer, 2008. – 546 p.

22. Bazaraa M.S. Nonlinear programming : theory and algorithms / M.S. Bazaraa, H.D. Sherali, C.M. Shetty. – 3-d edition. – USA, Hoboken, New Jersey : John Wiley & Sons Inc., 2006. – 853 p.

23. Sun W. Optimization theory and methods. Nonlinear programming / W. Sun, Y. X. Yuan. – USA, New York: Springer, 2006. – 687 p.

Навчальне видання

СЕВЕРИН Валерій Петрович

МЕТОДИ БАГАТОВИМІРНОЇ БЕЗУМОВНОЇ МІНІМІЗАЦІЇ

Навчальний посібник
з курсу «Методи оптимізації»

для студентів напрямків 6.040302 «Інформатика»,
6.040303 «Системний аналіз»

Російською мовою

Роботу до видання рекомендував *М. І. Безменов*
В авторській редакції

План 2013 р., п. 2

Підп. до друку 20.04.2013 р. Формат 60×84 1/16. Папір офісний.
Riso-друк. Гарнітура Таймс. Ум. друк. арк. 9,3. Наклад 100 прим.
Зам. № 126. Ціна договірна.

Видавець і виготовлювач
Видавничий центр НТУ «ХП»
вул. Фрунзе, 21, м. Харків-2, 61002

Свідоцтво суб'єкта видавничої справи ДК № 3657 від 24.12.2009 р.